Ph.D. Thesis
Doctor of Philosophy

DTU Compute
Department of Applied Mathematics and Computer Science

Deep Latent Variable Models

for Natural Language Processing

Valentin Liévin

Kongens Lyngby 2022

DTU Compute
Department of Applied Mathematics and Computer Science
Technical University of Denmark

Matematiktorvet
Building 303B
2800 Kongens Lyngby, Denmark
Phone +45 4525 3031
compute@compute.dtu.dk
www.compute.dtu.dk

Summary

Empowered by massively parallel computing and web-scale datasets, deep generative models have led to improvements in many artificial intelligence tasks. Latent variable models offer an elegant framework to augment generative algorithms with new capabilities. Nonetheless, in the field of natural language processing, it is unclear how best to combine latent variables with the powerful and omnipresent language models. In this thesis, we explore advanced optimization methods and the design of latent variable models to complete natural language processing tasks. The dissertation is organized into three parts.

In the first part, we present three architectures of latent variable language models. We discuss structured latent spaces which can be structured as large knowledge bases (e.g., Wikipedia) or characterized by a latent text generation process.

In the second part, we dive into the topic of variational inference and optimization. We present a novel gradient estimator for discrete latent variables named OVIS. We found that, even with OVIS, optimizing deep latent variable models remains challenging. We show, based on empirical data, that Rényi divergence variational inference can be applied to circumvent some of the learning issues.

In the third part, we focus on solving medical question answering and information retrieval tasks. First, we experiment with prompting pre-trained large language models (GPT-3) to generate step-by-step problem solutions. We report that, in many cases, GPT-3 can reason about challenging medical questions. Second, we introduce a new framework, dubbed VOD, for learning retrieval-augmented language models using variational inference. We apply VOD to optimize question-answering systems augmented with a large fraction of Wikipedia and report state-of-the-art performances on medical question-answering benchmarks. Last, we employ the resulting model for information retrieval tasks in the domain of rare disease diagnosis.

Resumé

Beriget med massiv parallel databehandling og store datasæt, har dybe generative modeller ført til forbedringer i mange opgaver indenfor kunstig intelligens. Latente variable modeller tilbyder et effektivt framework til at udvide generative algoritmer med nye kapaciteter. Ikke desto mindre er det uklart, hvordan man inden for sprogteknologien bedst kan kombinere latente variabler med de anvendelige og allestedsnærværende sprogmodeller. I denne afhandling udforsker vi advancerede optimeringsmetoder samt designet af latente variable modeller, som kan fuldføre opgaver indenfor sprogteknologi. Afhandlingen er opdelt i tre dele.

I første del gennemgår vi tre forskellige design af latente variable sprogmodeller. Vi diskuterer strukturerede latente rum, der kan struktureres som store vidensdatabaser (såsom f.eks. Wikipedia) eller karakteriseres ved en latent tekstgenereringsproces.

I anden del dykker vi ned i emnet variationel inferens og optimering. Vi præsenterer en ny gradientestimator for diskrete latente variabler ved navn OVIS. Vi har fundet, at det til stadighed forbliver vanskeligt at optimere dybe latente varible modeller, også med OVIS. Baseret på empiri, viser vi, at Rényi divergens variationel inferens kan anvendes til at omgå nogle af læringsproblemerne.

I den tredje og sidste del af afhandlingen, fokuserer vi på at løse medicinske *question-answering* og *information retrieval* opgaver. Først eksperimenterer vi med at få forudtrænede, store sprogmodeller (GPT-3) til at generere trin-for-trin problemløsninger. Vi rapporterer, at GPT-3 i mange tilfælde kan drage konklusioner vedrørende vanskelige medicinske spørgsmål. Efterfølgende introducerer vi et nyt framework, navngivet VOD, til at lære *retrieval-augmented* sprogmodeller ved hjælp af variationel inferens. Vi anvender VOD til at optimere *question-answering* systemer, beriget med Wikipedia artikler, og rapporterer state-of-the-art præstationer på medicinske *question-answering* benchmarks. Endelig, anvender vi den resulterende model for *information retrieval* opgaver inden for området: sjældne sygdomme.

Preface

This thesis was prepared at the Cognitive Systems section of the Department of Applied Mathematics and Computer Science (DTU Compute), Technical University of Denmark. It constitutes a partial fulfilment of the requirements for acquiring a PhD at the Technical University of Denmark.

This PhD project was financed by Google Deepmind through its PhD Scholarship Programme by DTU Compute. The project was supervised by Ole Winther (DTU Compute) and Casper Kaae Sønderby (co-founder of twig.energy). The PhD project was carried out at DTU during the period October 2018 - September 2022, except for a three-month leave of absence taken for an internship at FindZebra (summer 2021).

During the course of my PhD, I have worked on different topics in deep learning, variational inference, image modelling, language modelling, question answering and information retrieval. This research project resulted in four peer-reviewed publications and three preprints. This dissertation includes four of them.

Kongens Lyngby, September 14, 2022

Valentin Liévin

Contributions

Papers included in this thesis

A. **BIVA: A Very Deep Hierarchy of Latent Variables for Generative Modeling**
Maaløe, L., Fraccaro, M., Liévin, V., and Winther, O., 2019.
In *Advances in Neural Information Processing Systems 32*, pages 6551–62.

B. **Optimal Variance Control of the Score-Function Gradient Estimator for Importance-Weighted Bounds**
Liévin, V., Dittadi Andrea., Christensen A., and Winther O., 2020.
In *Advances in Neural Information Processing Systems 33*, pages 16591–602.

C. **Can Large Language Models Reason about Medical Questions?**
Liévin, V., Hother, E.C., Winther, O., 2022.
arXiv preprint arXiv:2207.08143.

D. **Variational Open-Domain Question Answering.**
Liévin, V., Motzfeldt, A.G., Jensen, I.R., Winther, O., 2022.
arXiv preprint arXiv:2210.06345.

Papers not included in this thesis

I. **Towards Hierarchical Discrete Variational Autoencoders**
Liévin, V., Dittadi A., Maaløe L., and Winther O., 2019.
In *Second Symposium on Advances in Approximate Bayesian Inference.*

II. **Image Super-Resolution With Deep Variational Autoencoders**
Chira, D., Haralampiev, I., Winther, O., Dittadi, A. and Liévin, V., 2022.
in *Advances in Image Manipulation workshop and challenges at ECCV 2022*, arXiv preprint arXiv:2203.09445.

III. **FindZebra online search delving into rare disease case reports using Natural Language Processing**
Liévin, V., Hansen, J.M., Lund, A., Elstein, D., Matthiesen, M.E., Elomaa, K., Zarakowska, K., Himmelhan, I., Botha, J., Borgeskov, H., Winther, O., 2022.
SSRN Preprint doi: 10.2139/ssrn.4198097.

Software

A. **BIVA for PyTorch**
`github.com/vlievin/biva-pytorch`

C. **OVIS**
`github.com/vlievin/ovis`

C. **Medical Reasoning using GPT-3**
`github.com/vlievin/medical-reasoning`

D. **Open-Domain Question Answering**
`github.com/vlievin/fz-openqa`

Acknowledgements

First and foremost I would like to thank my supervisor, Professor Ole Winther, for giving me the opportunity to begin this project. Thank you for your patience and for your unwavering support, from the first day to the last. Thank you for the invaluable scientific insights and for your exceptional contribution to OVIS.

I would also like to thank my collaborators and mentors who played a special role in framing my research interests; Casper Kaee Sønderby for prompting my interest in more statistically oriented methods and Lars Maaløe for giving me the opportunity to contribute to BIVA and for the continued mentoring.

I am very grateful to the people at FindZebra, Ole Winther, Mads Emil Matthiesen and Jonas Meinertz Hansen for welcoming me into the team, for your trust and for providing the perfect application for my research, this made this PhD project so much more rewarding.

I would also like to thank my close circle of colleagues, scientists and friends: Andrea Dittadi, Didriek Nielsen, Dimitris Kalatzis, Anders Christensen, Giorgio Giannone, and a special thanks to Jesper Wohlert, your honest feedback really helped me to improve.

I'd like to thank my family and friends in Denmark, France and elsewhere who've impacted the course of this PhD so positively, and in particular, thank you Thomas Hennequin, Karolina Molla Gazi, Quentin Tresontani, Laura Petranca and Gerard Montserrat Siso, Léo Averbuch and Charlotte Willems. And finally, I am deeply grateful to you Amalie Kubach for your unconditional and relentless support.

CONTENTS

Summary i

Resumé iii

Preface v

Contributions vii

Acknowledgements ix

Contents xi

1 Introduction 1
- 1 Thesis outline 2

I Background 5

2 Monte Carlo Methods 7
- 1 Importance Sampling 7
- 2 Gradient Estimation 11
- 3 Conclusion 15

3 Variational Inference 17
- 1 Latent Variable Models 17
- 2 Evidence Lower Bound 18
- 3 Importance-Weighted Bound 20
- 4 Variational Rényi Bound 21
- 5 Gradient Estimation 23
- 6 Conclusion 26

4 Deep Generative Models 27
- 1 Variational Autoencoders 27
- 2 Language Models 29
- 3 Conclusion 30

II Deep Latent Variable Models for Language Modelling 31

5 Bidirectional-Inference Variational Autoencoder 33
1 Architecture 34
2 Image modelling 35
3 Language modelling 35
4 Conclusion 37

6 Language as a Latent Variable 39
1 Latent Reasoning 39
2 Latent Information Retrieval 41
3 Conclusion 42

III Importance Weighted Bounds and Optimization 43

7 Optimal Variance of the Score Function Estimator 45
1 Background 46
2 OVIS 50
3 Empirical Validation 54
4 Conclusion 55
5 Bias of the STL Estimator for the Importance Weighted Bound 56

8 Taming Importance Weighted Bounds using Rényi Divergences 59
1 Navigating Between Variational Bounds 60
2 Empirical results 62
3 Conclusion 64

IV Medical Question Answering and Information Retrieval 65

9 Zero-Shot Medical Reasoning using Large Language Models 67
1 Background 68
2 Medical Reasoning with Large Language Models 70
3 Main Results 71
4 Conclusion 74

10 Learning to Search Wikipedia by Answering Medical Questions 77
1 Background 78
2 The Variational Open-Domain Framework 81
3 Main Results 84
4 Conclusion 86

11 Conclusion 87
1 Summary 87
2 Closing remarks 88

A BIVA: A Very Deep Hierarchy of Latent Variables for Generative Modeling 91

B Optimal Variance Control of the Score-Function Gradient Estimator for Importance-Weighted Bounds 117

C Can Large Language Models Reason about Medical Questions? 141

D Variational Open-Domain Question Answering 173

Bibliography 201

CHAPTER 1

Introduction

Computational science aims to understand and solve complex problems using advanced computing capabilities. Machine learning, a core discipline of computational science, has progressed immensely in the last decade. In particular, deep learning algorithms, empowered by the rise of massively parallel computing and large datasets, led to breakthroughs in various artificial intelligence tasks, such as computer vision and natural language processing. Machine learning algorithms are capable of harnessing the knowledge dormant in vast amounts of unstructured data (e.g., the internet, medical records, and physical measurements). The main motivation of this thesis is to apply the acquired knowledge to solve practical scientific and societal problems. For instance by providing better and more accessible healthcare thanks to the development of advanced diagnostic tools.

Deep generative models are a class of deep learning models that can learn complex distributions over high-dimensional data such as images or text. To learn robust and realistic generative processes, such models must acquire deep structural knowledge about the data domain from a large amount of unstructured training data. During the time of this thesis, generative models have progressed tremendously. They are now able to generate coherent text documents (Brown et al. 2020a; Radford et al. 2019a), to reason about problems using written language (Kojima et al. 2022) and to convert written instructions into photo-realistic images (Ramesh et al. 2022; Saharia et al. 2022) as well as videos (Singer et al. 2022).

This thesis aims to advance the state-of-the-art in deep generative modelling for natural language processing. In particular, we focus on latent variable models, a class of deep generative models which generative process is augmented with a set of unobserved auxiliary variables (*latent variables*). Although the beginning of this research was influenced by the literature in representation learning,[1] this work diverged from this path. This thesis focuses on the design and the optimization of advanced stochastic generative processes for text. In particular, we study variational inference and discrete latent variables, which, in some cases, encode complex actions such as searching through a knowledge base or generating hypotheses using written language. Ultimately, we are interested in developing methods that are of practical use; we apply latent variable language models to real-world natural language processing tasks:

---

[1] The beginning of this thesis was particularly inspired and motivated by "Generating Sentences from a Continuous Space" (Bowman et al. 2016)

answering medical questions and retrieving information to support the diagnostic of rare diseases.

1 Thesis outline

This dissertation is arranged into the main text that discusses and synthesizes the contributions of this thesis and an appendix section collecting the four included papers. It is recommended to use the main text as a basis to probe the different parts of the contributions and delve into the papers when technical derivations and further experimental details seem required.

The main body of this dissertation consists of four parts. The first part establishes the background knowledge supporting the remaining of the dissertation. This rest of the dissertation is organized around three main research questions:

Q1: How to design deep latent variable models for text data?
Q2: How to optimize latent variable models using variational inference?
Q3: How to apply latent variable models to natural language processing tasks?

Q1 will be treated in Part II, in which we present three types of latent variable models for natural language processing (Papers A, C and D). **Q2** will be treated in Part III, in which we discuss the topic of variational inference and optimization. We revisit OVIS, the gradient estimator for discrete latent variable models presented in Paper B and experiment with Rényi divergences to stabilize the optimization of deep latent variable models (Paper B and D). **Q3** will be treated in Part IV, in which we apply latent variable models to solve medical tasks: question answering and information retrieval (Papers C and D).

Part I Chapters 2, 3 and 4 introduce the relevant background for this dissertation. Chapter 2 reviews the basics of Monte Carlo methods, notably focusing on importance sampling and gradient estimation. Chapter 3 discusses latent variable models and variational inference, focusing on three types of multiple variational likelihood lower bounds. Chapter 4 presents the two classes of generative models: language models and Variational Autoencoders.

Part II (**Q1**) Chapters 5 and 6 provide an overview of the three types of latent variable models for text modelling. Chapter 5 presents BIVA, a deep hierarchical Variational Autoencoder (Paper A). Chapter 6 discusses two models which latent variable is text: a question-answering model with a latent *reasoning* process (Paper C) and an open-domain question-answering model, in which the latent space is a large collection of documents (Paper D).

Part III (**Q2**) In chapters 7 and 8, we review the contributions made in the domain of optimization latent variable models using importance weighted bounds. Chapter 7 revisits OVIS: a new class of gradient estimators for discrete latent variable models introduced in Paper B. We begin with a background section about variational inference for non-reparameterizable variational distributions and continue with an in-depth theoretical review of the OVIS estimators. We conclude with a brief review of the empirical results and a discussion of the limitations of our method. Chapter 8 discusses the use of Rényi divergence variational inference (Li and Turner 2016) to overcome some of the learning issues that arise when maximizing importance-weighted bounds. We present empirical results gathered from Papers B and D.

Part IV (**Q3**) Chapters 9 and 10 are concerned with the practical application of latent variable models for the task of medical question answering an information retrieval. Chapter 9 details a method for answering and reasoning about medical questions using large language models (Paper C). Chapter 10 concentrates on the task of open-domain question answering (Paper D). We present the Variational Open-Domain (VOD) framework; a framework for probabilistic and end-to-end learning of retrieval-augmented language models using variational inference. We review the main empirical results, covering the tasks of question-answering and information retrieval.

Part I

Background

CHAPTER 2

Monte Carlo Methods

Probability theory provides a framework for modelling many possible outcomes and quantifying knowledge; it is therefore a core component of machine learning systems. In this thesis, we study generative processes that are inherently stochastic and therefore are difficult to study because the analytical solution to the learning problem is often intractable. Monte Carlo methods, consisting of studying a system through repeated simulation, provide a framework to leverage advanced computing capabilities to understand and solve such complex problems.

In many applications, we are interested in estimating the weighted average of a function f given a probability distribution $p(\mathbf{z})$ where $\mathbf{z}$ is a random variable defined on a probability space $\mathcal{Z}$. This corresponds to the integral:

$$\mathcal{F} := \mathbb{E}_{p(\mathbf{z})}\left[f(\mathbf{z})\right] = \int_{\mathbf{z}\in\mathcal{Z}} p(\mathbf{z})f(\mathbf{z})d\mathbf{z} \ . \tag{2.1}$$

Nonetheless, evaluating the expected value might be impossible in practice, because the integration problem is intractable, or because the probability metric $p(\mathbf{z})$ is unknown. Monte Carlo simulation allows us to estimate the expected value using a set of N random samples $\mathbf{z}_1, \ldots, \mathbf{z}_N \sim p(\mathbf{z})$ drawn independently and with replacement:

$$\hat{\mathcal{F}}_N^{\mathrm{MC}} := \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{z}_i) \tag{2.2}$$

The estimator $\hat{\mathcal{F}}_N^{\mathrm{MC}}$ converges to the expected value $\mathbb{E}\left[f(\mathbf{z})\right]$ in the limit of large N, i.e., the estimation error $\hat{\mathcal{F}}_N^{\mathrm{MC}} - \mathbb{E}\left[f(\mathbf{z})\right]$ vanishes as $N \to \infty$ and has a variance

$$\mathrm{Var}\left[\hat{\mathcal{F}}_N^{\mathrm{MC}}\right] = \frac{1}{N}\sum_{i=1}^{N}\left(f(\mathbf{z}_i) - \mathbb{E}_{p(\mathbf{z})}\left[f(\mathbf{z})\right]\right)^2 \ . \tag{2.3}$$

This means that, given a sufficiently large number of samples, the integral 2.1 can be accurately approximated without knowing the analytical solution.

1 Importance Sampling

In other applications, it is impossible to sample from the distribution $p(\mathbf{z})$, even if the probability of each sample can be evaluated. In that setting, Importance sampling (IS) can be applied to evaluate the weighted average of $f(\mathbf{z})$ given samples drawn from a *proposal* distribution $q(\mathbf{z})$ satisfying $q(\mathbf{z}) > 0$ wherever $p(\mathbf{z}) > 0$. The importance sampling estimator is defined as

$$\hat{\mathcal{F}}_N^{\mathrm{IS}} := \frac{1}{N}\sum_{i=1}^{N} w(\mathbf{z}_i) f(\mathbf{z}_i) \tag{2.4}$$

with *importance weights* $w(\mathbf{z}) := \frac{p(\mathbf{z})}{q(\mathbf{z})}$ is an unbiased estimate of $\mathbb{E}_{p(\mathbf{z})}\left[f(\mathbf{z})\right]$ and consists of re-weighting the samples from $q(\mathbf{z})$ with the ratio of densities $w(\mathbf{z})$, as illustrated in Figure 2.1.

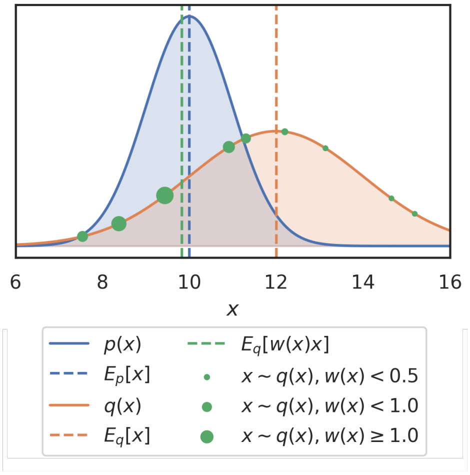


Figure 2.1: Importance Sampling. Estimating the mean of $p(x)$ using $N = 10$ samples from $q(x)$ and with importance weights defined as $w(x) = p(x)/q(x)$.

When applying an importance sampling estimator, the choice of the proposal distribution is essential. Although the variance of the estimator:

$$\mathrm{Var}[\hat{\mathcal{F}}_N^{\mathrm{IS}}] = \frac{1}{N}\mathbb{E}_{p(\mathbf{z})}\left[w^2(\mathbf{z})f^2(\mathbf{z}) - \mathcal{F}^2\right] \ , \tag{2.5}$$

scales as $\sim N^{-1}$ but depends on the choice of $q(\mathbf{z})$. The proposal distribution $q_\star(\mathbf{z})$ that minimizes the variance of the importance sampling estimator is (derivation below):

$$q_\star(\mathbf{z}) := \frac{p(\mathbf{z})|f(\mathbf{z})|}{\sum_{x\in\mathcal{Z}} p(\mathbf{z})|f(\mathbf{z})|} \ . \tag{2.6}$$

$$\begin{aligned}
\mathrm{Var}_{q(\mathbf{z})}\left[w(\mathbf{z})f(\mathbf{z})\right] &= \left(\int_{x\in\mathcal{Z}} q(\mathbf{z})\left(w(\mathbf{z})f(\mathbf{z})\right)^2 d\mathbf{z}\right)\left(\int_{x\in\mathcal{Z}} q(\mathbf{z})1^2 d\mathbf{z}\right) - \mathcal{F}^2 \\
&\geq \left(\int_{x\in\mathcal{Z}} q(\mathbf{z})w(\mathbf{z})|f(\mathbf{z})|d\mathbf{z}\right)^2 - \mathcal{F}^2 \quad \text{(Cauchy–Schwarz)} \\
&= Z\int_{x\in\mathcal{Z}} \frac{p^2(\mathbf{z})f^2(\mathbf{z})}{p(\mathbf{z})|f(\mathbf{z})|} d\mathbf{z} - \mathcal{F}^2 \\
&= \int_{x\in\mathcal{Z}} q_\star(\mathbf{z})w_\star^2(\mathbf{z})f^2(\mathbf{z}) - \mathcal{F}^2 d\mathbf{z} \\
&= \mathrm{Var}_{q_\star(\mathbf{z})}\left[w_\star(\mathbf{z})|f(\mathbf{z})|\right] \ .
\end{aligned}$$

1.1 Self-normalized importance sampling

The standard importance sampling estimator is unbiased. Nonetheless, in practice, we might prefer to use a biased self-normalized version of the estimator because the standard estimator (i) might suffer from high variance (ii) is intractable because $p(\mathbf{z})$ is only known up to a constant. *self-normalized importance sampling* (SNIS) is defined based on the un-normalized weights $\bar{w}(\mathbf{z}) := Z \cdot w(\mathbf{z})$ (where Z is a constant):

$$\hat{\mathcal{F}}_N^{\mathrm{SNIS}} := \frac{1}{N} \sum_{i=1}^{N} \tilde{w}(\mathbf{z}_i) f(\mathbf{z}_i) \quad \text{where} \quad \tilde{w}(\mathbf{z}_i) := \frac{\bar{w}(\mathbf{z}_i)}{\sum_{i'=1}^{N} \bar{w}(\mathbf{z}_{i'})} \,. \tag{2.8}$$

The estimator $\hat{\mathcal{F}}_N^{\mathrm{SNIS}}$ is consistent (i.e. it converges to the true expected value with probability one) as the denominator $\sum_{i'=1}^{N} \bar{w}(\mathbf{z}_{i'})$ converges to $\mathbb{E}_{q(\mathbf{z}')}\left[\bar{w}(\mathbf{z}')\right] = Z$. The bias of $\hat{\mathcal{F}}_N^{\mathrm{SNIS}}$ scales as $\sim N^{-1}$ and the variance is approximately (Kong 1992):

$$\mathrm{Var}_{q(\mathbf{z})}\left[\hat{\mathcal{F}}_N^{\mathrm{SNIS}}\right] \approx \mathrm{Var}_{p(\mathbf{z})}\left[\hat{\mathcal{F}}_N^{\mathrm{MC}}\right] \left(1 + \mathrm{Var}_{q(\mathbf{z})}\left[\bar{w}(\mathbf{z})\right]\right) \,. \tag{2.9}$$

Thus, we can expect the variance of the self-normalized importance sampling estimate to be as low as the variance of the standard Monte Carlo estimate 2.2 as long as the variance of the importance weights $\mathrm{Var}_{q(\mathbf{z})}\left[\bar{w}(\mathbf{z})\right]$ is sufficiently low.

1.2 Effective sample size

Equation 2.9 quantifies the variance of the importance sampling estimate relative to the variance of a standard Monte Carlo estimate (with samples drawn from $p(\mathbf{z})$). Kong 1992 suggested using the above result to define the Effective Sample Size (ESS) to diagnose self-normalized importance sampling estimates. The ESS is defined as:

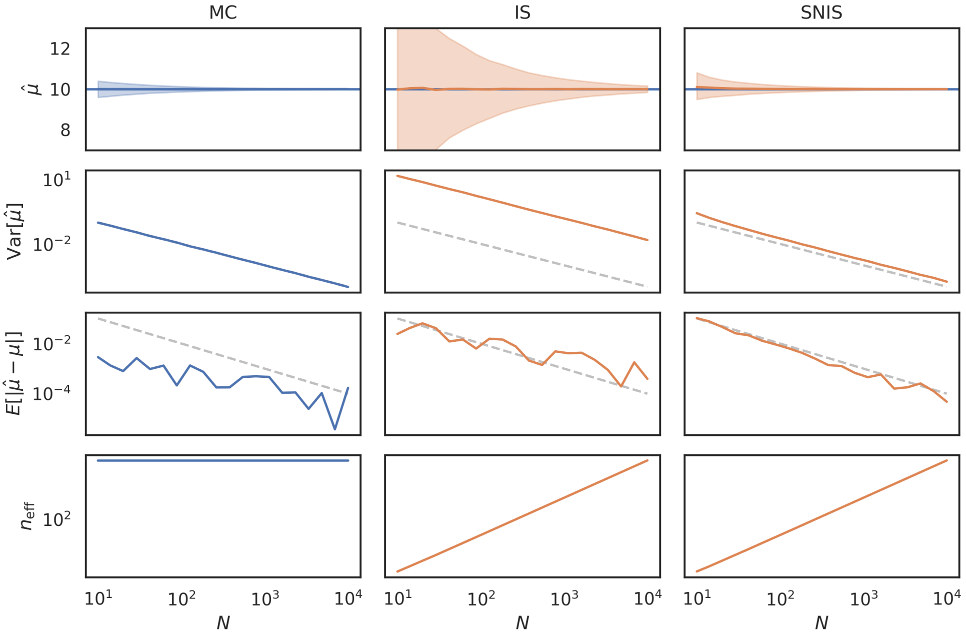
$$n_{\mathrm{eff}} := \frac{N}{1 + \mathrm{Var}_{q(\mathbf{z}),N}\left[\bar{w}(\mathbf{z})\right]} \approx \frac{\left(\sum_{i=1}^{N} \bar{w}(\mathbf{z}_i)\right)^2}{\sum_{i=1}^{N} \bar{w}^2(\mathbf{z}_i)} \,. \tag{2.10}$$

In practice, The value of the effective sample size n_{eff} corresponds to the number of samples drawn from $q(\mathbf{z})$ that are assigned with a weight $w(\mathbf{z})$ which value is significantly large to affect the importance sampling estimate; it measures how many sample from $q(\mathbf{z})$ are *effectively* used to estimate $\mathbb{E}_{p(\mathbf{z})}\left[f(\mathbf{z})\right]$.

Limitations and alternative ESS The effective sample size might fail whenever all samples from $q(\mathbf{z})$ are weighted equally and $p(\mathbf{z}) \approx 0$. In that case, the effective sample size is large, but the estimate is likely inaccurate; one might then turn to use the variance of $\hat{\mathcal{F}}_N^{\mathrm{SNIS}}$ as a secondary diagnostic.

Nonetheless, The optimal proposal defined in Equation 2.6 is proportional $p(\mathbf{z})|f(\mathbf{z})|$ and n_{eff} is independent of the value $f(\mathbf{z})$. Alternatively, the effective sample size can be defined with weights $\bar{w}(\mathbf{z}) = \frac{|f(\mathbf{z})|p(\mathbf{z})}{q(\mathbf{z})}$, which is the definition adopted in this thesis.[1] In Figure 2.2, we report the bias, variance and effective sample size of three estimators: standard MC, IS and SNIS.

Figure 2.2: Variance, bias and effective sample size. Estimating the mean of $p(z) = \mathcal{N}(10, 1^2)$ using standard Monte Carlo (MC), importance sampling (IS) and self-normalized importance sampling (SNIS) with $q(z) = \mathcal{N}(12, 2^2)$. We report trends for 10000 draws for each value of $N = 1 \ldots 10000$ samples.



1.3 Priority sampling

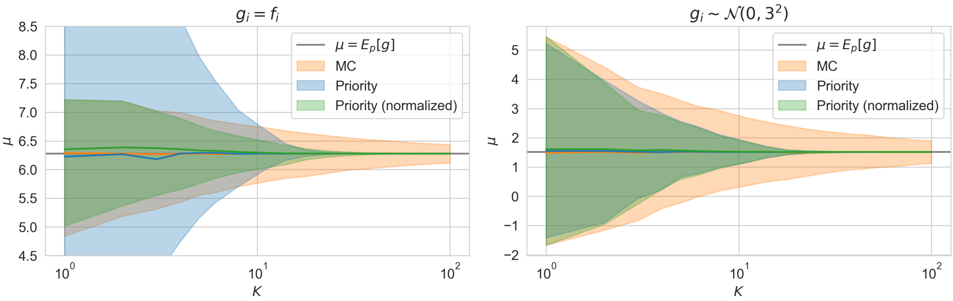
In the case of a finite sampling space, estimating the weighted average via standard Monte Carlo might be inefficient because each sample can be drawn multiple times. *Priority sampling* (Duffield, Lund, and Thorup 2007) allows estimating the weighted average using samples drawn without replacement.

Given a sample space $\mathcal{Z}$ of size N with probabilities $p_1, \ldots, p_N$ and a function with values $f_1, \ldots, f_N$, priority sampling allows estimating the weighted average $\sum_{i=1}^{N} p_i f_i$ using a subset of $K < N$ samples. Using a sequence of random weights $u_1, \ldots, u_n \sim \mathrm{Uniform}(0, 1]$, we define a priority key p_i / u_i for each value in the sampling space

---

[1] See Nowozin 2015 for a great introduction to effective sample size and Owen 2013, Chapter 9, for more information about the diagnostic of importance samplers and the effective sample size.

Figure 2.3: Priority sampling. Estimation of $\mathcal{F} = \mathbb{E}_p[g]$ with probabilities $p_i := \sum_{i=1}^{N} \exp f_i / \sum_{j=1} \exp f_j$, with $f_i \sim \mathcal{N}(0, 3^2)$ and with $N = 100$. We apply Monte Carlo (sampling with replacement), priority sampling and priority sampling with self-normalized weights (sampling without replacement). We use $g_i = f_i$ in the left side of the plot, and values $g_i \sim \mathcal{N}(0, 3^2)$ sampled independently of f_i in the right side. In both cases, self-normalized priority sampling (green) converges faster than the two other estimators.



and let τ be the $K + 1$-th largest key. The value of the threshold τ defines a set of K samples $\mathbb{S} = \{i \in [1, N] \mid p_i/u_i > \tau\}$. Priority sampling can be interpreted as an importance sampling estimator with weights $s_i := \max(p_i, \tau)$; this gives us an unbiased estimate of the weighted average:

$$\mathbb{E}_{p(u_1,\ldots,u_N)}\left[\sum_{j\in\mathbb{S}} s_j f_j\right] = \sum_{i=1}^{N} p_i f_i \ . \tag{2.11}$$

The priority sampling estimate 2.11 is an importance sampling estimate and therefore it is often beneficial to apply self-normalization (Section 1.1) with weights $\tilde{s}_i = s_i \ \sum_{j\in\mathbb{S}} s_j f_j$. In figure 2.3, we illustrate priority sampling and self-normalize priority sampling. We refer to Vieira 2017 for more information about priority sampling, including the proof of unbiasedness for Equation 2.11.

2 Gradient Estimation

Automatic differentiation software like Tensorflow (Abadi et al. 2016), PyTorch (Paszke et al. 2019) and JAX (Frostig, Johnson, and Leary 2018) allow differentiating deep neural networks and optimizing them via backpropagation. In the case of Monte Carlo objectives such as Equation 2.1, the computation graph includes one or many sampling steps which are not trivially differentiable. In this section, we present different strategies to obtain differentiable Monte Carlo estimates of the gradient of Monte Carlo objectives with respect to the parameter of the probability distribution.

We consider a distribution $p_\phi(\mathbf{z})$ parameterized by $\phi \in \Phi$, and $f_\theta(\mathbf{z})$ a function of $\mathbf{z}$ parameterized by $\theta \in \Theta$, both defined on a probability space $\mathcal{Z}$, this defines Monte Carlo objective

$$\mathcal{F}(\theta, \phi) := \mathbb{E}_{p_\phi(\mathbf{z})}\left[f_\theta(\mathbf{z})\right] \tag{2.12}$$

as a function of both parameters. In this section, we seek to estimate the gradient of the stochastic objective 2.12 w.r.t. the parameter ϕ, defined here as:

$$\eta := \nabla_\phi \mathcal{F}(\theta, \phi)\ . \tag{2.13}$$

The gradient is a vector of size $|\phi|$, where each dimension corresponds to the derivative of $\mathcal{F}(\theta, \phi)$ w.r.t. the corresponding parameter dimension. In modern deep learning applications, both the probability space $\mathcal{Z}$ and the parameter space Φ can be composed of a large number of dimensions, making the gradient of the integral $\mathbb{E}_{p_\phi(\mathbf{z})}\left[f_\theta(\mathbf{z})\right]$ w.r.t. the parameter θ intractable.

Monte Carlo methods provide a solution to estimate the gradients by simulating the objective function. Monte Carlo gradient estimators can be categorized into three classes (Mohamed et al. 2019): (i) the *pathwise estimator*, (ii) the *score-function estimator* and (iii) the *measure-valued gradient estimator*. In the following section, we discuss the pathwise and score function estimators and refer to Mohamed et al. 2019 for a comprehensive review of gradient estimation using Monte Carlo methods.

2.1 Pathwise estimator

The pathwise estimator consists in exploiting the structure of the function $f_\theta(\mathbf{x})$ to provide an estimate of the gradient. Given a base distribution $p_\phi(\mathbf{z})$, if a deterministic *sampling path* $g(\boldsymbol{\epsilon}; \phi)$ and a base distribution $p(\boldsymbol{\epsilon})$ can be found such that the sampling processes $\mathbf{z} \sim p_\phi(\mathbf{z})$ and $\mathbf{z} = g_\phi(\boldsymbol{\epsilon}), \boldsymbol{\epsilon} \sim p(\boldsymbol{\epsilon})$ are equivalent. Then, by the *law of the unconscious statistician* (LOTUS) (Grimmett, Grimmett, and Stirzaker 2001), the objective $\mathcal{F}(\theta, \phi)$ can be expressed as an expectation over the distribution $p(\boldsymbol{\epsilon})$:

$$\mathcal{F}(\theta, \phi) = \mathbb{E}_{p(\boldsymbol{\epsilon})}\left[f_\theta(g_\phi(\boldsymbol{\epsilon}))\right]\ . \tag{2.14}$$

Thanks to this re-parameterization of the sampling path, the parameter ϕ is made independent of the sampling distribution and can be *pushed in* the cost function:

$$\nabla \phi \mathcal{F}(\theta, \phi) = \mathbb{E}_{p(\boldsymbol{\epsilon})}\left[\nabla_\phi f_\theta(g_\phi(\boldsymbol{\epsilon}))\right]\ . \tag{2.15}$$

Equation 2.15 can be estimated via Monte Carlo; this defines the pathwise estimator:

$$\bar{\eta}_K^{\text{pathwise}} := \frac{1}{K} \sum_{i=1}^{N} \nabla_\phi f_\theta(g_\phi(\boldsymbol{\epsilon}_i)), \quad \boldsymbol{\epsilon}_1, \ldots, \boldsymbol{\epsilon}_K \sim p(\boldsymbol{\epsilon})\ . \tag{2.16}$$

The estimator 2.16 can be evaluated using automatic differentiation packages, allowing for end-to-end optimization of stochastic objectives. It has been popularized in the field of deep learning as the *reparameterization trick* (Kingma and Welling 2014a), and as *stochastic backpropagation* (Rezende, Mohamed, and Wierstra 2014) and has since then been widely adopted by the community. When applicable, the estimator provides a low variance estimator in most settings thanks to the structural information provided by the differentiation of the sampling path. Nonetheless, the estimator cannot be directly applied to discrete distributions. In that case, we might turn to other estimators, such as the score function estimator.

2.2 Score function estimator

In statistics, the *score function* is defined as the gradient of the log probability $\nabla_\phi \log p_\phi(\mathbf{z})$ w.r.t. the parameter ϕ. The score function estimator consists in utilizing the identity $\nabla_\phi \log p_\phi(\mathbf{z}) = \nabla_\theta p_\phi(\mathbf{z}) / p_\phi(\mathbf{z})$ to express the gradient of objective function as an expected value of the gradient $\nabla_\phi \log p_\phi(\mathbf{z})$ weighted by the value $f_\theta(\mathbf{z})$:

$$\begin{aligned}
\nabla \phi \mathcal{F}(\theta, \phi) =& \nabla_\phi \int_{\mathbf{z}} p_\phi(\mathbf{z}) f_\theta(\mathbf{z}) d\mathbf{z} && (2.17a)\\
=& \int_{\mathbf{z}} \nabla_\phi p_\phi(\mathbf{z}) f_\theta(\mathbf{z}) d\mathbf{z} && (2.17b)\\
=& \int_{\mathbf{z}} p_\phi(\mathbf{z}) \nabla_\phi \log p_\phi(\mathbf{z}) f_\theta(\mathbf{z}) d\mathbf{z} && (2.17c)\\
=& \mathbb{E}_{p_\phi(\mathbf{z})} \left[f_\theta(\mathbf{z}) \nabla_\phi \log p_\phi(\mathbf{z}) \right] \ . && (2.17d)
\end{aligned}$$

Estimating Equation 2.17d via Monte Carlo yields the score function gradient estimator Kleijnen and Rubinstein 1996, also known as the *REINFORCE* estimator (Williams 1992):

$$\bar{\eta}_K^{\text{reinforce}} := \frac{1}{K} \sum_{i=1}^{N} f_\theta(\mathbf{z}) \nabla_\phi \log p_\phi(\mathbf{z}_i), \quad \mathbf{z}_1, \ldots, \mathbf{z}_K \sim p_\phi(\mathbf{z}) \ . \tag{2.18}$$

The score function estimator is generally applicable, and doesn't require the function f_θ to be differentiable, nor does it require $p_\phi(\mathbf{z})$ to be reparameterizable. This makes REINFORCE a great candidate for optimizing black-box systems, stochastic control flows, or discrete latent variable models. This estimator is also infamous for suffering from high variance. Nonetheless, it is possible to reduce variance by coupling the estimator with control variates (Section 2.4).

2.3 Assessing gradient estimators

Bias and variance Monte Carlo estimates of the gradient $\eta = \nabla_\phi \mathcal{F}(\theta, \phi)$ are multivariate random variables and their distributions can be evaluated by studying their moments: the expected value $\mathbb{E}[\bar{\eta}_K]$, with associated bias $\|\mathbb{E}[\bar{\eta}_K] - \eta\|$, and the covariance $\operatorname{Cov}[\bar{\eta}_K]$. In practice, this is however often simpler and informative enough to study the bias and variance of each parameter component separately.

Unbiased estimators are often preferred to their biased counterparts because unbiasedness guarantees that increasing the number of samples will result in a more accurate estimation of the true gradients. Two unbiased estimators have equal expected values. In that case, studying the variance is sufficient to compare both estimators.

SNR Nonetheless, as explored in Chapter 8, it is sometimes necessary to trade unbiasedness for lower-variance estimates. In that case, comparing biased estimators requires further knowledge of the task. Ultimately evaluating the end performances on the downstream task is the ideal solution, but intermediate gradient statistics can still be helpful. The signal-to-noise ratio (SNR) (Rainforth et al. 2018) gauges the quality of a gradient estimate by measuring the amount of noise relative to the magnitude of the expected value (Rainforth et al. 2018). The SNR is defined for the ith component of the parameter vector θ_i as

$$\mathrm{SNR}\left[\bar{\eta}_K(i)\right] = \frac{\left|\mathbb{E}\left[\bar{\eta}_K(i)\right]\right|}{\sqrt{\mathrm{Var}\left[\bar{\eta}_K(i)\right]}} \ . \tag{2.19}$$

2.4 Control variates

The computation requirements grow linearly with the number of samples K. When working under a constrained computational budget, reducing the variance of a gradient estimator is key to improving learning. The variance of a K-samples gradient estimator η_K can be reduce using a *control variate* β which value correlates with η such that:

$$\mathbb{E}\left[\beta\right] = 0 \quad \text{and} \quad \mathrm{Var}\left[\eta_K - \beta\right] \leq \mathrm{Var}\left[\eta_K\right] \ . \tag{2.20}$$

Although variance control can be applied to pathwise estimator (Roeder, Wu, and Duvenaud 2017; Tucker et al. 2019), variance control is more often utilized in combination with the score function estimator because its structure can more easily be exploited. The expected value of the gradient of the score function is zero, therefore for a constant baseline c (e.g., $c = \mathbb{E}[f_\theta(\mathbf{z})]$), we have:

$$\mathbb{E}_{p_\phi(\mathbf{z})}\left[c \cdot \nabla_\phi \log p_\phi(\mathbf{z})\right] = c \cdot \mathbb{E}_{p_\phi(\mathbf{z})}\left[\frac{\nabla_\phi p_\phi(\mathbf{z})}{p_\phi(\mathbf{z})}\right] = c \cdot \nabla_\phi \mathbb{E}_{p_\phi(\mathbf{z})}\left[1\right] = 0 \ . \tag{2.21}$$

When estimating the estimator $\bar{\eta}_K^{\text{reinforce}}$ (Equation 2.18) using $K > 1$, it is possible to recycle the set of samples $\mathbf{z}_{1:K} := \{\mathbf{z}_1, \dots \mathbf{z}_K\}$ and their prefactor values $f_\theta(\mathbf{z}_1), \dots f_\theta(\mathbf{z}_K)$ in the design of baselines (Mnih and Rezende 2016). Indeed, given independently drawn samples, the value of a function $\hat{f}_\theta(\mathbf{z}_{-i})$, that takes the set of samples $\mathbf{z}_{-i} = \mathbf{z}_{1:K} \backslash \{\mathbf{z}_i\}$ as argument, is independent of the variable $\mathbf{z}$, and therefore independent of all functions of $\mathbf{z}$. It follows that

$$\begin{aligned}\mathbb{E}_{p_\phi(\mathbf{z}_{1:K})}\left[\hat{f}_\theta(\mathbf{z}_{-i})\nabla_\phi \log p_\phi(\mathbf{z}_i)\right] = \mathbb{E}_{p_\phi(\mathbf{z}_{1:K})}\left[\hat{f}_\theta(\mathbf{z}_{-i})\right] & \overbrace{\mathbb{E}_{p_\phi(\mathbf{z}_{1:K})}\left[\nabla_\phi \log p_\phi(\mathbf{z}_i)\right]}^{=0\,(\text{Equation } 2.21)} \\ & + \underbrace{\mathrm{Cov}_{p_\phi(\mathbf{z}_{1:K})}\left[\hat{f}_\theta(\mathbf{z}_{-i}), \nabla_\phi \log p_\phi(\mathbf{z}_i)\right]}_{=0\,(\text{independence of } \mathbf{z}_i \text{ and } \mathbf{z}_{-i})}\end{aligned} \tag{2.22}$$

and therefore a multi-sample baseline of the following form has zero expectation:

$$\beta(\mathbf{z}_{1:K}) := \sum_{i=1}^{K} \hat{f}_\theta(\mathbf{z}_{-i})\nabla_\phi \log p_\phi(\mathbf{z}_i), \quad \mathbf{z}_{1:K} \sim p_\phi(\mathbf{z}_{1:K}) \ . \tag{2.23}$$

3 Conclusion

We reviewed the basics of evaluating Monte Carlo objectives and their gradients. Importance sampling is a method for evaluating stochastic objectives given samples drawn from auxiliary distributions, which is especially useful when samples from the target distribution cannot be obtained. We presented two solutions to estimating the gradients of Monte Carlo objectives: the pathwise estimator, a low-variance gradient estimator applicable to reparameterizable distributions and the score function estimator, which is more widely applicable but suffers from high variance unless effective control variates can be designed. In Chapter 7, we will present a novel control variate for the score function estimator.

Monte Carlo methods offer an elegant framework to trade computation for better estimation accuracy. Nonetheless, in the case of complex stochastic processes, computation resources are constrained, and standard Monte Carlo estimates have high variance. Importance sampling is a solution to this problem. Nonetheless identifying an adequate proposal distribution is still required. This will be the topic of the next chapter, in which we will discuss latent variable models and variational inference.

CHAPTER 3

Variational Inference

In this chapter, we study the problem of inference in *latent variable* models. Latent variables are unobserved variables and latent variable models defined a generative process defining a directed path from the *latent* space to the *observation* space. Inference consists in identifying the reverse process: quantifying the properties of an unobserved variable given an observed variable. Analytical solutions to this problem are generally intractable. Instead of solving learning and inference of graphical models exactly,

> "*variational methods convert a complex problem into a simpler problem, where the simpler problem is generally characterized by a decoupling of the degrees of freedom in the original problem. This decoupling is achieved via an expansion of the problem to include additional parameters, known as variational parameters, that must be fit to the problem at hand.*" — Jordan et al. 1999.

In the following sections, we begin with an introduction to latent variable models before discussing the topic of variational inference. We review multiple variational objectives, which all can be applied to learn latent variable models via maximum likelihood estimation. We conclude with deriving gradient estimates of the variational objectives using the Monte Carlo methods presented in Chapter 2.

1 Latent Variable Models

We consider an observable random variable $\mathbf{x}$ defined on a space $\mathcal{X}$ and an unobserved random variable $\mathbf{z}$ defined on a latent space $\mathcal{Z}$ with prior distribution $p_\theta(\mathbf{z})$. We define the decoder of the generative model as a conditional distribution $p_\theta(\mathbf{x} \mid \mathbf{z})$. This results in a latent variable generative model characterized by the *joint* distribution:

$$p_\theta(\mathbf{x}, \mathbf{z}) := p_\theta(\mathbf{x} \mid \mathbf{z})\, p_\theta(\mathbf{z}) \ . \tag{3.1}$$

Although the variable $\mathbf{z}$ is not observed, prior knowledge about the target generative process $\mathbf{x} \sim p_{\text{data}}(\mathbf{x})$ can be encoded with an adequate choice of space $\mathcal{Z}$, prior $p_\theta(\mathbf{z})$ and decoder $p_\theta(\mathbf{z} \mid \mathbf{x})$.

Marginal Likelihood The joint distribution describes a generative process controlled by a parameter θ, that allows first sampling a latent representation and then generating a corresponding observation. Even with a simple prior distribution $p_\theta(z)$, the coupling described by the joint distribution $p_\theta(\mathbf{x}, \mathbf{z})$ results in modelling more complex densities in the observation space $\mathcal{X}$ via marginalization of the latent variable $\mathbf{z} \in \mathcal{Z}$. This corresponds to the *marginal likelihood*:

$$p_\theta(\mathbf{x}) := \int_{\mathcal{Z}} p_\theta(\mathbf{x}, \mathbf{z}) d\mathbf{z} = \int_{\mathcal{Z}} p_\theta(\mathbf{z}) p_\theta(\mathbf{x} \mid \mathbf{z}) d\mathbf{z} \ . \tag{3.2}$$

The marginal likelihood can be utilized to optimize the parameter θ via maximum likelihood estimation. Marginalizing over the prior $p_\theta(z)$ is often impossible. Nonetheless, we recognize in Equation 3.2 a Monte Carlo objective, for which solutions for approximate integration in Chapter 2.

Approximate Inference Importance sampling (Section 1) can be applied to estimate the marginal likelihood 3.2 using sample from a proposal distribution $q(\mathbf{z})$. The distribution minimizing the variance of an importance sampler is (Equation 2.6):

$$q_\star(\mathbf{z}) := \frac{p_\theta(\mathbf{x} \mid \mathbf{z}) p_\theta(\mathbf{z})}{\int_{\mathcal{Z}} p_\theta(\mathbf{x} \mid \mathbf{z}) p_\theta(\mathbf{z}) d\mathbf{z}} = p_\theta(\mathbf{z} \mid \mathbf{x}) \ . \tag{3.3}$$

This corresponds to the *posterior* distribution $p_\theta(\mathbf{z} \mid \mathbf{x})$, which connects the generative model and the marginal likelihood via *Bayes' rule* ($p_\theta(\mathbf{x} \mid \mathbf{z}) p_\theta(\mathbf{z}) = p_\theta(\mathbf{z} \mid \mathbf{x}) p_\theta(\mathbf{x})$). Nonetheless, the posterior distribution depends on the marginal likelihood and thus is itself intractable. Variational inference consists of augmenting the problem with an additional parameter ϕ, dubbed *variational parameter*, which parameterizes an *approximate posterior* $q_\phi(\mathbf{z} \mid \mathbf{x})$. In this thesis, the approximate posterior is implemented using deep neural networks and is therefore referred to as *inference network*.

In the following sections, we detail multiple *variational* objectives that can be utilized to learn the inference network jointly with the generative model. In the remaining Chapters, we will study these objectives and apply them to learn deep latent variable models.

Alternative approaches, not studied in this thesis, exist and are viable alternatives. They include: Monte Carlo estimation of the posterior such as the *Expectation-Maximization* (EM) algorithm (Dempster, Laird, and Rubin 1977), designing the generative model using invertible transformations (flows, Dinh, Sohl-Dickstein, and Bengio 2017) or constraining the approximate posterior to a simple and fixed distribution (diffusion models, Ho, Jain, and Abbeel 2020).

2 Evidence Lower Bound

In variational inference, we seek an objective that allows for maximizing the marginal likelihood 3.2 and aligning the approximate posterior $q_\phi(\mathbf{z} \mid \mathbf{x})$ with the true posterior $p_\theta(\mathbf{z} \mid \mathbf{x})$. We begin by introducing a statistical distance to measure the divergence between two distributions.

Shannon entropy The entropy, denoted $\mathcal{H}$, was introduced by Shannon (1948) to quantify the predictability information of a random process; in other words, its information content. The Shannon entropy of a variable x with distribution $p(x)$ is a non-negative value defined as

$$\mathcal{H}(p) := -\sum_{x \in \mathcal{X}} p(x) \log p(x) \ . \tag{3.4}$$

The Kullback-Leibler (KL) divergence, or relative entropy (Kullback and Leibler 1951), measures the difference in information content between two distributions. For two distribution $p(x)$ and $q(x)$ both defined on $\mathcal{X}$ such that $p(x) = 0$ where $q(x) = 0$, the relative entropy from p to q defined as

$$\mathcal{D}_{\mathrm{KL}}(q \,\|\, p) = -\sum_{x \in \mathcal{X}} q(x) \log \frac{p(x)}{q(x)} \ . \tag{3.5}$$

The KL divergence defines a valid *statistical divergence*: it is (i) non-negative: $\mathcal{D}_{\mathrm{KL}}(q \| p) \geq 0$, (ii) additive: for two distributions $q(x) = q_1(x) q_2(x)$ and $p(x) = p_1(x) p_2(x)$, $\mathcal{D}_{\mathrm{KL}}(q \| p) = \mathcal{D}_{\mathrm{KL}}(q_1 \| p_1) + \mathcal{D}_{\mathrm{KL}}(q_2 \| p_2)$, and (iii) is equal to zero if and only if $p = q$.

Variational objective The optimal variational distribution $q_\phi(\mathbf{z} \mid \mathbf{x})$ can be chosen as the distribution with the variational parameter that minimizes the KL divergence:

$$\begin{aligned} \mathcal{D}_{\mathrm{KL}}\left(q_\phi(\mathbf{z} \mid \mathbf{z}) \,||\, p_\theta(\mathbf{z} \mid \mathbf{x})\right) &:= \int_{\mathcal{Z}} q_\phi(\mathbf{z} \mid \mathbf{x}) \left(\log q_\phi(\mathbf{z} \mid \mathbf{x}) - \log p_\theta(\mathbf{z} \mid \mathbf{x})\right) d\mathbf{z} && (3.6\text{a}) \\ &= \log p_\theta(\mathbf{x}) - \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z} \mid \mathbf{x})} \right] \ . && (3.6\text{b}) \end{aligned}$$

Removing the non-negative divergence from Equation 3.6 yields a log-likelihood lower bound. This is the *evidence lower bound* (ELBO), denoted $\mathcal{L}$ and defined as:

$$\mathcal{L}(\mathbf{x}) := \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z} \mid \mathbf{x})} \right] \leq \log p_\theta(\mathbf{x}) \ . \tag{3.7}$$

The ELBO satisfies both of the learning requirements: it allows maximizing the marginal likelihood (lower bound) and allows minimizing a divergence between the true posterior and the approximate posterior observing that $\nabla_\phi \mathcal{L}(\mathbf{x}) = -\nabla_\phi \mathcal{D}_{\mathrm{KL}}\left(q_\phi(\mathbf{z} \mid \mathbf{z}) \,||\, p_\theta(\mathbf{z} \mid \mathbf{x})\right)$. Furthermore, it can be decomposed into two terms, which are insightful from an auto-encoding perspective:

$$\mathcal{L}(\mathbf{x}) = \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log p_\theta(\mathbf{x} \mid \mathbf{z})\right]}_{\text{(A) } \textit{reconstruction}} - \underbrace{\mathcal{D}_{\mathrm{KL}}(q_\phi(\mathbf{z} \mid \mathbf{x}) \,\|\, p_\theta(\mathbf{z}))}_{\text{(B) } \textit{regularization}} \ . \tag{3.8}$$

3 Importance-Weighted Bound

As a direct consequence of Equation 3.6, the ELBO aligns with the marginal likelihood if and only if $q_\phi(\mathbf{z}|\mathbf{x}) = p_\theta(\mathbf{z}|\mathbf{x})$ and this difference equals the value of the divergence $\mathcal{D}_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{z})\,||\,p_\theta(\mathbf{z}|\mathbf{x})\right)$. In some cases, this divergence might be large and learning by maximizing the variational bound might be ineffective (See Cremer, Li, and Duvenaud 2018 for a study on this topic).

Importance sampling In Section 1, we discussed that importance sampling provides unbiased estimates, using $q_\phi(\mathbf{z} \mid \mathbf{x})$ as proposal distribution, the log marginal likelihood can be expressed as an expectation over $q_\phi(\mathbf{z} \mid \mathbf{x})$:

$$\log p_\theta(\mathbf{x}) = \log \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\frac{p_\theta(\mathbf{x},\mathbf{z})}{q_\phi(\mathbf{z} \mid \mathbf{x})}\right] . \tag{3.9}$$

Because of the log function, a K-sample Monte Carlo estimate of Equation 3.9 is generally biased (Jensen's inequality), nonetheless, it converges to the log marginal likelihood in the limit $K \to \infty$, in contrast with the ELBO.

Tighter variational bound A K-sample Monte Carlo estimate of Equation 3.9 defines the *importance-weighted bound* (Burda, Grosse, and Salakhutdinov 2016):

$$\mathcal{L}^K(\mathbf{x}) := \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})}\left[\log \frac{1}{K}\sum_{i=1}^{K} w_{\theta,\phi}(\mathbf{x},\mathbf{z}_i)\right] \tag{3.10}$$

where $\mathbf{z}_{1:K} := \{\mathbf{z}_1, \ldots, \mathbf{z}_K\}$ is a set of Monte Carlo samples drawn from $q_\phi(\mathbf{z} \mid \mathbf{x})$ and with importance weight $w_{\theta,\phi}(\mathbf{x},\mathbf{z}) := p_\theta(\mathbf{x},\mathbf{z})/q_\phi(\mathbf{z},\mathbf{x})$. The importance-weighted bound non-decreasing with K and remains a lower bound of the marginal likelihood for all values $K \geq 1$:

$$\mathcal{L}(\mathbf{x}) \leq \mathcal{L}^K(\mathbf{x}) \leq \mathcal{L}^{K+1}(\mathbf{x}) \leq \log p_\theta(\mathbf{x}) \ . \tag{3.11}$$

Therefore, using importance-weighted bound guarantees estimating the marginal likelihood more tightly as $K \to \infty$, and corresponds to the ELBO in $K = 1$ (see Figure 3.1).

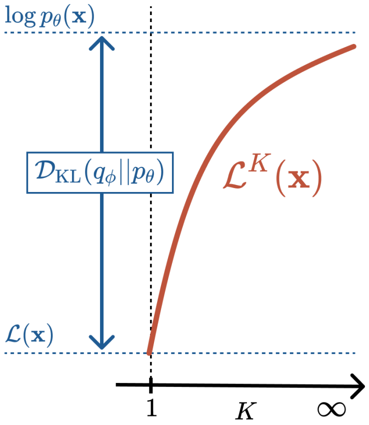


Figure 3.1: Beyond the ELBO: tighter estimation of the marginal likelihood using $K > 1$ Monte Carlo samples using the importance weighted bound $\mathcal{L}^K(\mathbf{x})$.

4 Variational Rényi Bound

Rényi entropy In Section 2, we introduced Shannon entropy as a measure of the information content inherent to a probability distribution. Rényi (1961) generalized the notion of entropy to a wider class of measures. Given a parameter $\alpha \geq 0$, the Rényi entropy is defined as

$$\mathcal{H}_\alpha(p) := \frac{1}{1-\alpha} \sum_{x \in \mathcal{X}} p^\alpha(x) \tag{3.12}$$

for $\alpha \neq 1$ and its definition is extended in $\alpha = 1$ by continuity as the Shannon entropy:

$$\mathcal{H}_{\alpha=1}(p) := \lim_{\alpha \to 1} \mathcal{H}_\alpha(p) = \mathcal{H}(p) \ . \tag{3.13}$$

The Rényi entropy is a measure of information for $\alpha > 0$ but is constant for $\alpha = 0$. In figure 3.2, we illustrate the Rényi entropy for $\alpha \geq 0$ based on Bernoulli distribution with parameter $\theta \in [0, 1]$.

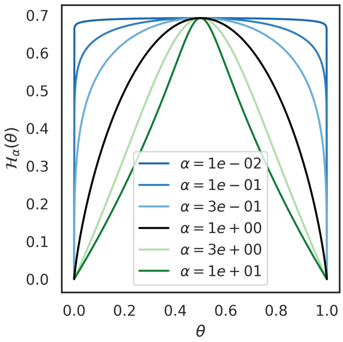


Figure 3.2: Rényi entropy $\mathcal{H}_\alpha$ measured for a Bernoulli distribution with parameter $\theta \in [0, 1]$. $\mathcal{H}_{\alpha=1}$ corresponds to the Shannon entropy. Different values of α lead to different uncertainty quantification profiles.

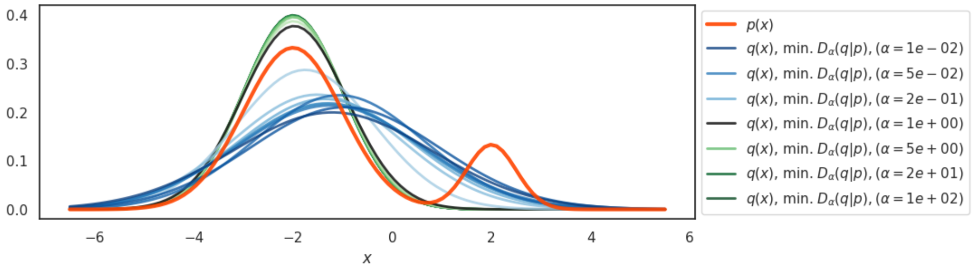


Figure 3.3: Fitting a multi-modal distribution $p(z)$ (in **red**) with a Gaussian model $q_\phi(z)$ by minimizing the Rényi divergence $\mathcal{D}_\alpha(q_\phi \| p)$ for values of $\alpha > 0$ (**blue**: $\alpha < 1$ and **green**: $\alpha > 1$). Using $\alpha \to 0$ leads to a *mean seeking* behaviour, $\alpha \geq 1$ leads to *seeking the mode.*

Rényi divergence Similarly to Shannon entropy (Equation 3.4), the Rényi entropy can be applied to defined a statistical divergence, the *Rényi divergence*, defined as

$$\mathcal{D}_\alpha\left(q \,\|\, p\right) := \frac{1}{\alpha - 1} \log \int_{\mathcal{X}} q(x) \left(\frac{p(x)}{q(x)}\right)^{1-\alpha} dx \tag{3.14}$$

for $\alpha \neq 1$ and extended by continuity in $\alpha = 1$ where it corresponds to the KL divergence (Equation 3.5). The Rényi divergence is a statistical divergence for $\alpha > 0$ and wherever $|\mathcal{D}_\alpha\left(q \| p\right)| < \infty$. As illustrated in Figure 3.3, minimizing the Rényi divergence between two distributions leads to different solutions depending on the choice of α. For $\alpha \geq 1$, minimizing the Rényi divergence leads to matching only the mode of the distribution. For $\alpha \to 0$, it leads to covering the whole target distribution.

Rényi variational bound The evidence lower bound arises from attempting to minimize the KL divergence from the true posterior to the approximate posterior (Section 2). Another variational bound can be obtained using the Rényi divergence $\mathcal{D}_\alpha\left(q_\phi(\mathbf{z} \mid \mathbf{x}) \| p_\theta(\mathbf{z} \mid \mathbf{x})\right)$ instead of the KL divergence. The resulting *Rényi variational bound* (RVB) is defined as (Li and Turner 2016):

$$\mathcal{L}_\alpha(\mathbf{x}) := \frac{1}{1-\alpha} \log \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[w_{\theta,\phi}^{1-\alpha}(\mathbf{x},\mathbf{z})\right] . \tag{3.15}$$

for $\alpha \neq 1$ and defined in $\alpha = 1$ by continuity as $\mathcal{L}_{\alpha=1}(\mathbf{x}) := \lim_{\alpha\to1} \mathcal{L}_\alpha(\mathbf{x}) = \mathcal{L}(\mathbf{x})$, where $\mathcal{L}(\mathbf{x})$ is the standard evidence lower bound (Equation 3.7). The RVB is a continuous and non-decreasing function of α and coincides with the marginal log-likelihood in $\alpha = 0$. Summarizing the above information, the RVB connects the ELBO and the marginal likelihood through the four following equations:

$$\mathcal{L}_{\alpha=1}(\mathbf{x}) = \mathcal{L}(\mathbf{x}) \qquad \mathcal{L}_{\alpha=0}(\mathbf{x}) = \log p_\theta(\mathbf{x}) \tag{3.16a}$$

$$\mathcal{L}_{\alpha\geq0}(\mathbf{x}) \leq \log p_\theta(\mathbf{x}) \qquad \mathcal{L}_{\alpha\leq0}(\mathbf{x}) \geq \log p_\theta(\mathbf{x})\,. \tag{3.16b}$$

Importance weighted Rényi bound In general, the RVB has no closed-form solution. a K-sample estimate of the RVB yields another bound, the *importance-weighted Rényi bound* (IWRB), which is defined for $\alpha \neq 1$ as:

$$\mathcal{L}_\alpha^K(\mathbf{x}) := \frac{1}{1-\alpha} \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})}\left[\log \frac{1}{K} \sum_{i=1}^{K} w_{\theta,\phi}^{1-\alpha}(\mathbf{x},\mathbf{z}_i)\right] \tag{3.17}$$

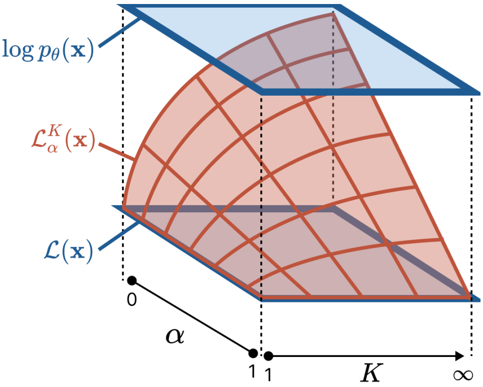

$\log p_\theta(\mathbf{x})$
$\mathcal{L}_\alpha^K(\mathbf{x})$
$\mathcal{L}(\mathbf{x})$
0
α
1
1
K
∞

Figure 3.4: The importance weighted Rényi bound (IWRB) as a function of the parameters $\alpha \in [0, 1]$ and $K \geq 1$.

The definition of the IWR bound is extended in $\alpha = 1$ by continuity with the IW bound (Equation 3.10). The bound satisfies the following properties:

- $\mathcal{L}(\mathbf{x}) = \mathcal{L}_\alpha^{K=1}$,
- $\lim_{K\to\infty} \mathcal{L}_\alpha^K(\mathbf{x}) = \mathcal{L}_\alpha(\mathbf{x})$,
- non-decreasing in K for fixed $\alpha \leq 1$,
- non-increasing in K for fixed $\alpha \geq 1$,
- continuous and non-increasing in α

The IWRB is a useful tool to navigate between the ELBO and the log-likelihood by controlling the value of K and $\alpha \in [0, 1]$, we illustrate the corresponding 2D surface in Figure 3.4).

5 Gradient Estimation

In this section, we derive the gradients of the IWRB with respect to each parameter (θ and ϕ) and present corresponding Monte Carlo estimators. The IWRB includes the IWB as a special case ($\alpha = 0$), but the gradients of the ELBO need to be derived separately, as the IWRB is extended in $\alpha = 1$ by continuity.

5.1 IWRB: parameter θ

The gradient of the IWRB with respect to θ can be expressed as:

$$\nabla_\theta \mathcal{L}_\alpha^K(\mathbf{x}) = \frac{1}{1-\alpha} \nabla_\theta \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})} \left[\log \frac{1}{K} \sum_{i=1}^K w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, \mathbf{z}_i) \right] \tag{3.18a}$$

$$= \frac{1}{1-\alpha} \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})} \left[\nabla_\theta \log \frac{1}{K} \sum_{i=1}^K w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, \mathbf{z}_i) \right] \tag{3.18b}$$

$$= \frac{1}{1-\alpha} \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})} \left[\frac{\nabla_\theta \sum_{i=1}^K w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, \mathbf{z}_i)}{\sum_{i=1}^K w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, \mathbf{z}_i)} \right] \tag{3.18c}$$

$$= \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})} \left[\sum_{i=1}^K \widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{x}, \mathbf{z}_i) \nabla_\theta \log p_\theta(\mathbf{x}, \mathbf{z}_i) \right] \tag{3.18d}$$

where in the last line we used the score function identity $\nabla \log p_\theta(\mathbf{x}, \mathbf{z}) = \frac{\nabla_\theta p_\theta(\mathbf{x},\mathbf{z})}{p_\theta(\mathbf{x},\mathbf{z})}$ and concluded using the expression of the self-normalizing weights defined as:

$$\widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{x}, \mathbf{z}) := \frac{w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, \mathbf{z})}{\sum_{i'=1}^K w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, \mathbf{z}_{i'})} \,. \tag{3.19}$$

The gradient $\nabla_\theta \mathcal{L}_\alpha^K(\mathbf{x})$ is an expected value with respect to the approximate posterior. We can be approximated using the following K-sample Monte Carlo estimator

$$\bar{\eta}_{\theta,N} := \sum_{i=1}^K \widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{x}, \mathbf{z}_i) \nabla_\theta \log p_\theta(\mathbf{x}, \mathbf{z}_i), \quad \mathbf{z}_{1:K} \sim q_\phi(\mathbf{z}_{1:K}|\mathbf{x}) \,. \tag{3.20}$$

5.2 IWRB: parameter ϕ

Pathwise estimator In the case of a reparameterizable distribution $q_\phi(\mathbf{z}|\mathbf{x})$, for instance a Gaussian distribution with diagonal covariance matrix and location parameters conditioned on $\mathbf{x}$, there is a base distribution $p(\boldsymbol{\epsilon})$ and a function g_ϕ such that the process $\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})$ and $\boldsymbol{\epsilon} \sim p(\boldsymbol{\epsilon}), \mathbf{z} = g_\phi(\mathbf{z})$ are equivalent. Therefore, the pathwise estimator (Section 2) can be applied to the gradient with respect to the variational parameter. The pathwise gradient of the IWRB is:

$$
\begin{aligned}
\nabla_\phi \mathcal{L}_\alpha^K(\mathbf{x}) =& \frac{1}{1-\alpha} \nabla_\phi \mathbb{E}_{p(\boldsymbol{\epsilon}_{1:K})} \left[\log \frac{1}{K} \sum_{i=1}^{K} w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, g_\phi(\boldsymbol{\epsilon}_i)) \right] && \text{(3.21a)} \\
=& \frac{1}{1-\alpha} \mathbb{E}_{p(\boldsymbol{\epsilon}_{1:K})} \left[\nabla_\phi \log \frac{1}{K} \sum_{i=1}^{K} w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, g_\phi(\boldsymbol{\epsilon}_i)) \right] && \text{(3.21b)} \\
=& \frac{1}{1-\alpha} \mathbb{E}_{p(\boldsymbol{\epsilon}_{1:K})} \left[\frac{\nabla_\theta \sum_{i=1}^{K} w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, g_\phi(\boldsymbol{\epsilon}_i))}{\sum_{i=1}^{K} w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, g_\phi(\boldsymbol{\epsilon}_i))} \right] && \text{(3.21c)} \\
=& -\mathbb{E}_{p(\boldsymbol{\epsilon}_{1:K})} \left[\sum_{i=1}^{K} \widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{x}, \mathbf{z}_i) \nabla_\theta \log q_\phi(g_\phi(\boldsymbol{\epsilon}_i) \mid \mathbf{x}) \right] , && \text{(3.21d)}
\end{aligned}
$$

which can be estimated using the following Monte Carlo estimator:

$$
\bar{\eta}_{\phi,N}^{\text{pathwise}} := -\sum_{i=1}^{K} \widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{x}, \mathbf{z}_i) \nabla_\phi \log q_\phi(g_\phi(\boldsymbol{\epsilon}_i) \mid \mathbf{x}), \quad \boldsymbol{\epsilon}_{1:K} \sim p(\boldsymbol{\epsilon}_{1:K}) \ . \tag{3.22}
$$

Score function estimator The gradient of the IWRB w.r.t. the parameter ϕ of the variational distribution is (see derivation in the box below):

$$
\nabla_\phi \mathcal{L}_\alpha^K(\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}_{1:K} \mid \mathbf{x})} \left[\sum_{i=1}^{K} \left(\log \hat{Z}_\alpha(\mathbf{x} \mid \mathbf{z}_{1:K}) - \widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{x}, \mathbf{z}_i) \right) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x}) \right] \tag{3.23}
$$

where we introduced a marginal likelihood estimate, unbiased for $\alpha = 0$, defined as

$$
\hat{Z}_\alpha(\mathbf{x} \mid \mathbf{z}_{1:K}) := \left(\frac{1}{K} \sum_{i=1}^{K} w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, \mathbf{z}_i) \right)^{1/1-\alpha} . \tag{3.24}
$$

A K-sample Monte Carlo estimate of Equation 3.23 yields a score function estimator (see Section 2):

$$
\bar{\eta}_{\phi,K}^{\text{reinforce}} := \sum_{i=1}^{K} \left(d_{\alpha,i}(\mathbf{x}, \mathbf{z}_{1:K}) - c_\alpha(\mathbf{x}, \mathbf{z}_{-i}) \right) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x}), \quad \mathbf{z}_{1:K} \sim q_\phi(\mathbf{z} \mid \mathbf{x}) \ . \tag{3.25}
$$

where

$$
d_{\alpha,i}(\mathbf{x}, \mathbf{z}_{1:K}) := \log \hat{Z}_\alpha(\mathbf{x} \mid \mathbf{z}_{1:K}) - \widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{x}, \mathbf{z}_i) \tag{3.26}
$$

and where $c_\alpha(\mathbf{x}, \mathbf{z}_{-i})$ is a baseline function defining a control variate:

$$
\beta_\alpha(\mathbf{z}_{1:K}) := \sum_{i=1}^{K} c_\alpha(\mathbf{x}, \mathbf{z}_{-i}) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x}), \quad \mathbb{E}_{q_\phi(\mathbf{z}_{1:K} \mid \mathbf{x})} \left[\beta_\alpha(\mathbf{z}_{1:K}) \right] = 0 \ . \tag{3.27}
$$

$$\begin{aligned}
\nabla_\phi \mathcal{L}^K_\alpha(\mathbf{x}) =& \frac{1}{1-\alpha}\nabla_\phi \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})}\left[\log \frac{1}{K}\sum_{i=1}^{K} w^{1-\alpha}_{\theta,\phi}(\mathbf{x},\mathbf{z}_i)\right] \\
=& \int_{\mathbf{z}} \frac{1}{1-\alpha} \log \frac{1}{K}\sum_{i=1}^{K} w^{1-\alpha}_{\theta,\phi}(\mathbf{x},\mathbf{z}_i) \nabla_\phi q_\phi(\mathbf{z}_{1:K} \mid \mathbf{x}) d\mathbf{z} \\
&+ \int_{\mathbf{z}} \frac{1}{1-\alpha} q_\phi(\mathbf{z}_{1:K} \mid \mathbf{x}) \nabla_\phi \log \frac{1}{K}\sum_{i=1}^{K} w^{1-\alpha}_{\theta,\phi}(\mathbf{x},\mathbf{z}_i) d\mathbf{z} \\
=& \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})}\left[\hat{Z}_\alpha(\mathbf{x} \mid \mathbf{z}_{1:K}) \nabla_\phi \log q_\phi(\mathbf{z}_{1:K} \mid \mathbf{x})\right] \\
&+ \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})}\left[\frac{1}{1-\alpha}\sum_{i=1}^{K} \frac{\nabla_\phi w^{1-\alpha}_{\theta,\phi}(\mathbf{x},\mathbf{z}_i)}{\sum_{i'=1}^{K} w^{1-\alpha}_{\theta,\phi}(\mathbf{x},\mathbf{z}_{i'})}\right] \\
=& \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})}\left[\sum_{i=1}^{K} \left(\log \hat{Z}_\alpha(\mathbf{x} \mid \mathbf{z}_{1:K}) - \widetilde{w^{1-\alpha}_{\theta,\phi}}(\mathbf{x},\mathbf{z}_i)\right) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x})\right]
\end{aligned}$$

(Using the identity $\nabla_\phi w^{1-\alpha}_{\theta,\phi}(\mathbf{x},\mathbf{z}_i) = (1-\alpha) w^{1-\alpha}_{\theta,\phi}(\mathbf{x},\mathbf{z}_i) \nabla_\phi \log w_{\theta,\phi}(\mathbf{x},\mathbf{z}_i)$.)

control variates Control variates can be designed in any way as long as the control variate $\beta_\alpha(\mathbf{z}_{1:K})$ has zero expectation or a positive effect on the learning dynamics. Previous research focused on designing baseline functions $c_\alpha(\mathbf{x},\mathbf{z}_{-i})$ that approximate the prefactor term $d_{\alpha,i}(\mathbf{x},\mathbf{z}_{1:K})$ as accurately as possible.

In the variational inference literature, we find baselines designed using an auxiliary neural network (NVIL, Mnih and Gregor 2014a), baselines that recycle the values of $w_{\theta,\phi}(\mathbf{x},\mathbf{z}_1), \ldots, w_{\theta,\phi}(\mathbf{x},\mathbf{z}_K)$ (VIMCO, (Mnih and Rezende 2016)) and control variates that exploit the gradient of the pathwise estimator (REBAR, Tucker et al. 2017; RELAX, Grathwohl et al. 2018). In Chapter 7 (Paper B), we introduce OVIS, a new estimator that extends VIMCO with a more optimal baseline.

5.3 ELBO

The gradient of the ELBO with respect to the parameter θ is:

$$\nabla_\theta \mathcal{L}(\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}_{1:K})}\left[\frac{1}{K}\sum_{i=1}^{K} \nabla_\theta \log w_{\theta,\phi}(\mathbf{x},\mathbf{z}_i)\right] . \tag{3.29}$$

The gradient of the ELBO with respect to the parameter ϕ is:

$$\begin{aligned}
\nabla_\phi \mathcal{L}(\mathbf{x}) =& \nabla_\phi \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log w_{\theta,\phi}(\mathbf{x},\mathbf{z})\right] && (3.30a)\\
=& \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\left(\log w_{\theta,\phi}(\mathbf{x},\mathbf{z}) - 1\right) \nabla_\phi \log q_\phi(\mathbf{z} \mid \mathbf{x})\right] && (3.30b)\\
=& \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\log w_{\theta,\phi}(\mathbf{x},\mathbf{z}) \nabla_\phi \log q_\phi(\mathbf{z} \mid \mathbf{x})\right] && (3.30c)\\
=& \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})} \left[\frac{1}{K} \sum_{i=1}^{K} \log w_{\theta,\phi}(\mathbf{x},\mathbf{z}_i) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x})\right] . && (3.30d)
\end{aligned}$$

In the above, we use the identity $\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\nabla_\phi \log q_\phi(\mathbf{z}|\mathbf{x})\right] = 0$. we spare the reader from the expressions of the corresponding Monte Carlo estimates, as the process to obtain them is similar to the one followed in the case of the IWRB.

6 Conclusion

We have introduced variational inference, a method that simplifies the learning problem by augmenting it with additional parameters. The additional parameter (variational parameter) parameterizes an approximate posterior $q_\phi(\mathbf{z} \mid \mathbf{x})$, which can be used to define lower bounds of the log marginal likelihood. The resulting variational objectives enable joint training of the generative model and the approximate posterior.

We presented four variations of variational objectives: the evidence lower bound (ELBO), the importance-weighted bound (IWB), the Rényi variational bound (RVB) and its K-sample estimate, the importance-weighted Rényi bound (IWRB). Last, we derived gradients of the ELBO and gradient estimators for the IWRB, which include the IWB as a special case.

In Chapter 7, we will present a novel class of control variates for the score function estimator of the IWB. In Chapter 8, we will discuss the use of the RVB to optimize deep latent variable models more effectively.

CHAPTER 4

Deep Generative Models

In this chapter, we introduce the two main classes of text generative models studied in the remaining chapters: the Variational Autoencoder and the language models, considering both the autoregressive and masked variants.[1] This chapter does not cover other generative models such as flows (Dinh, Sohl-Dickstein, and Bengio 2017; Kingma and Dhariwal 2018), energy-based models (Grathwohl et al. 2020; LeCun et al. 2006) and diffusion models (Ho, Jain, and Abbeel 2020; Kingma, Salimans, Poole, et al. 2021). In the following sections, we consider the problem of modelling an observable variable $\mathbf{x}$ (e.g., image, text) with a distribution $p_\theta(\mathbf{x})$. In all cases, we assume the model to be parameterized by deep neural networks.[2]

1 Variational Autoencoders

For a tutorial on Variational Autoencoders, see
`github.com/DeepLearningDTU/02456-deep-learning-with-PyTorch/blob/master/7_Unsupervised`
`7.2-EXE-variational-autoencoder.ipynb`

The Variational Autoencoder (VAE, Kingma and Welling 2014b; Rezende, Mohamed, and Wierstra 2014) is a latent variable model parameterized by deep neural networks. It consists of a generative model $p_\theta(\mathbf{x}, \mathbf{z}) = p_\theta(\mathbf{z})\, p_\theta(\mathbf{z} \mid \mathbf{x})$ and an *inference network* $q_\phi(\mathbf{z} \mid \mathbf{x})$. The VAE is designed for variational inference (see chapter 3); the inference network q_ϕ corresponds to an approximate posterior with variational parameter ϕ. VAEs are trained by maximizing a variational bound (Section 2, 3 and 4).

Illustration Let's consider the variable $\mathbf{x}$ representing binary data (e.g., MNIST). We choose a d-dimensional diagonal Gaussian prior, parameterize the posterior with neural networks g_ϕ^μ and g_ϕ^σ and parameterize the decoder with a neural network f_θ,

---

[1] Although masked language models like BERT (Devlin et al. 2019) are not – strictly speaking – generative models, we include them here to simplify the structure of the thesis. We refer the reader to Hoogeboom et al. 2021 for a discussion on the interpretation of masked language models as an instance of autoregressive diffusion model.

[2] See Goodfellow, Bengio, and Courville 2016 for an introduction to deep learning.

all with output dimension d. In that case, a simple VAE can be expressed as:

$$p_\theta(\mathbf{x} \mid \mathbf{z}) = \text{Bernoulli}(f_\theta(\mathbf{z})) \tag{4.1a}$$

$$p_\theta(\mathbf{z}) = \mathcal{N}(0, I) \tag{4.1b}$$

$$q_\phi(\mathbf{z} \mid \mathbf{x}) = \mathcal{N}(g^\mu_\phi(\mathbf{x}), \exp g^\sigma_\phi(\mathbf{x}))\,. \tag{4.1c}$$

where the variational distribution $q_\phi(\mathbf{z} \mid \mathbf{x})$ can is reparameterizable because the sampling process $\mathbf{z} \sim \mathcal{N}(g^\mu_\phi(\mathbf{x}), \exp g^\sigma_\phi(\mathbf{x}))$ is equivalent to $\mathbf{z} = g^\mu_\phi(\mathbf{x}) + \epsilon \odot \exp g^\sigma_\phi(\mathbf{x}), \epsilon \sim \mathcal{N}(0, I))$. The resulting VAE can be optimized using the evidence lower bound (Section 2):

$$\mathcal{L}(\mathbf{x}) = \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x} \mid \mathbf{z})\right]}_{\text{(A) } \textit{reconstruction}} - \underbrace{\mathcal{D}_{\text{KL}}(q_\phi(\mathbf{z} \mid \mathbf{x}) \,\|\, p_\theta(\mathbf{z}))}_{\text{(B) } \textit{regularization}}\,. \tag{4.2}$$

which can be optimized end-to-end using the pathwise estimator (Section 2.1). At evaluation time, the marginal likelihood can be approximated using the importance-weighted bound (Equation 3.10) with a sample budget as large as possible.

Designing VAEs VAEs come in many shapes and forms.[3] We can identify three design dimensions: the choice of the posterior distribution $q_\phi(\mathbf{z} \mid \mathbf{x})$, the choice of the prior distribution $p_\theta(\mathbf{z})$, the choice of the observation model $p_\theta(\mathbf{x} \mid \mathbf{z})$. Each model can be designed with an arbitrary factorization (e.g., autoregressive decoder, hierarchical prior, autoregressive prior, autoregressive posterior) and various base distributions can be employed within each of the three components (e.g., Gaussian distributions, flows or categorical distributions). In Chapter 5, we present an example of VAE with an autoregressive decoder and which prior factorizes as a deep hierarchy of latent variables, each of which is modelled by a base Gaussian distribution.

Training objectives The variational objectives presented in Chapter 3 are all valid candidates to train VAEs, although the evidence lower bound is more typical. Variational objectives are sometimes altered to circumvent optimization problems or to promote learning useful latent representations.

Posterior collapse The evidence lower bound (Equation 4.2) is composed of (A) a reconstruction term and (B) a regularization term that pushes the approximate posterior to align with the prior. In practice, the regularization term might collapse to zero, leading to the latent variables being unused. *Posterior collapse* can be avoided by weakening the decoder (Bowman et al. 2016; Chen et al. 2017b) or by modifying the training objective. *Deterministic warm-up* (Bowman et al. 2016; Sønderby et al. 2016) consists in altering the evidence lower bound with a parameter β:

$$\mathcal{L}_\beta(\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x} \mid \mathbf{z})\right] - \beta \cdot \mathcal{D}_{\text{KL}}(q_\phi(\mathbf{z} \mid \mathbf{x}) \,\|\, p_\theta(\mathbf{z}))\,, \tag{4.3}$$

---

[3] We recommend Kingma and Welling 2019 for a complete background on VAEs

which can be increased from zero to one during early training such as to reduce the effect of the regularizing term. Alternatively, the objective can be modified with an amount of λ *free bits* (Kingma et al. 2016), which approximately corresponds to:

$$\mathcal{L}_\lambda(\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}\left[\log p_\theta(\mathbf{x} \mid \mathbf{z})\right] - \max\{\lambda, \mathcal{D}_{\mathrm{KL}}(q_\phi(\mathbf{z} \mid \mathbf{x}) \,\|\, p_\theta(\mathbf{z}))\}\,. \tag{4.4}$$

Representation learning In some modelling scenarios, it is possible to learn latent representations that are aligned with high-level features such as a class label. Ideally, these features are compact and easily re-usable for downstream tasks. Notably, Higgins et al. 2017 showed that increasing the regularization term in Equation 4.3 by scaling the parameter β up can promote learning of *disentangled* representations. Nonetheless, such features are difficult to learn without additional supervision. In practice, semi-supervised learning (Kingma et al. 2014) allows aligning the latent representations with known features in a more predictable way.

2 Language Models

For a tutorial on language modelling, RNNs and Transformers, see `github.com/DeepLearningDTU/02456-deep-learning-with-PyTorch/blob/master/5_Transformers/5_1_EXE_deep_learning_with_transformers.ipynb`

In this section, the variable $\mathbf{x}$ an arbitrary piece of text encoded at the character-level, word-level, or sub-word level (Sennrich, Haddow, and Birch 2016; Song et al. 2021). After tokenization, the variable $\mathbf{x}$ decomposes into a sequence of T tokens $x_1, \ldots, x_T$. A *language model* corresponds to the joint distribution $p_\theta(\mathbf{x})$. Although simple language models can be defined using n-gram models, in this section, we discuss those parameterized by deep neural networks.

Autoregressive language models Most of the autoregressive language models adopt a left-to-right factorization of the tokens:

$$p_\theta(\mathbf{x}) = \prod_{t=1}^{T} p_\theta(x_t \mid \mathbf{x}_{<t})\,, \tag{4.5}$$

where $\mathbf{x}_{<t} = x_1, \ldots, x_{t-1}$. Each transition model $p_\theta(x_t \mid \mathbf{x}_{<t})$ is parameterized by a neural network f_θ such that:

$$p_\theta(x_t \mid \mathbf{x}_{<t}) = \mathrm{Cat}\left(x_t \mid f_\theta(\mathbf{x}_{<t})\right)\,. \tag{4.6}$$

For instance, f_θ can be implemented using a recurrent neural network such as an LSTM (Hochreiter and Schmidhuber 1997) or using a Transformer (Vaswani et al. 2017).

The above factorization can be extended to permutations of the indices $1, \ldots, T$; as long as the causal dependencies are enforced. In the case of a Transformer-based model, the dependencies are enforced by masking the attention mechanism. Order-agnostic language models can be obtained by sampling random orderings during training, as explored in Yang et al. 2019 and in Hoogeboom et al. 2021.

Masked language models Alternatively, other methods such as BERT (Devlin et al. 2019) trade the strict autoregressive dependencies for a set of two self-supervised objectives. The first objective corresponds to a masked language modelling task characterized by the conditional:

$$p_\theta(\mathbf{x}_\mathbf{m} \mid \mathbf{x}_{-\mathbf{m}}) = \prod_{j \in \mathbf{m}} p_\theta(x_j \mid \mathbf{x}_{-\mathbf{m}}) \,, \tag{4.7}$$

where $\mathbf{m}$ corresponds to the indices of the masked tokens to be predicted and $-\mathbf{m}$ represents all the other indices. Masked language models are typically implemented using masked self-attention. Similarly to the autoregressive language model, we can express the conditional using a deep neural network f_θ:

$$p_\theta(x_j \mid \mathbf{x}_{-\mathbf{m}}) = \mathrm{Cat}\left(x_j \mid f_\theta(\mathbf{x}_{-\mathbf{m}})\right) \,. \tag{4.8}$$

The second task consists of predicting whether two sentences appear next to one another in the training corpus. This promotes BERT not only to operate at the token level but also to model the relationship between two sentences. We refer to the original paper for a more complete description of the training algorithm (Devlin et al. 2019).

Large scale pre-training Both types of language models can be trained using large amounts of unlabelled text data. Autoregressive language models such as GPT(Brown et al. 2020a; Radford et al. 2018; Radford et al. 2019b) are train to maximize the marginal likelihood $p_\theta(\mathbf{x})$. Masked language models like BERT (Devlin et al. 2019) are trained to maximize the conditional $p_\theta(\mathbf{x}_\mathbf{m} \mid \mathbf{x}_{-\mathbf{m}})$ and to solve the next sentence prediction task. In both cases, the underlying neural network f_θ learns to extract syntactic and semantic dependencies from the context ($\mathbf{x}_{<t}$ or $\mathbf{x}_{-\mathbf{m}}$). These resulting contextual representations, trained on large amounts of data, often align with reusable language features and can be applied to solve downstream tasks.

3 Conclusion

In this chapter, we have briefly introduced the Variational Autoencoder and covered the basics of language modelling. This concludes the part I of this thesis. In the next Chapter, we will discuss how to combine variational autoencoders and language models.

Part II

Deep Latent Variable Models for Language Modelling

CHAPTER 5

Bidirectional-Inference Variational Autoencoder

This chapter cites one of the contributions in this thesis:

Paper A: Maaløe et al. 2019 "BIVA: A Very Deep Hierarchy of Latent Variables for Generative Modeling"

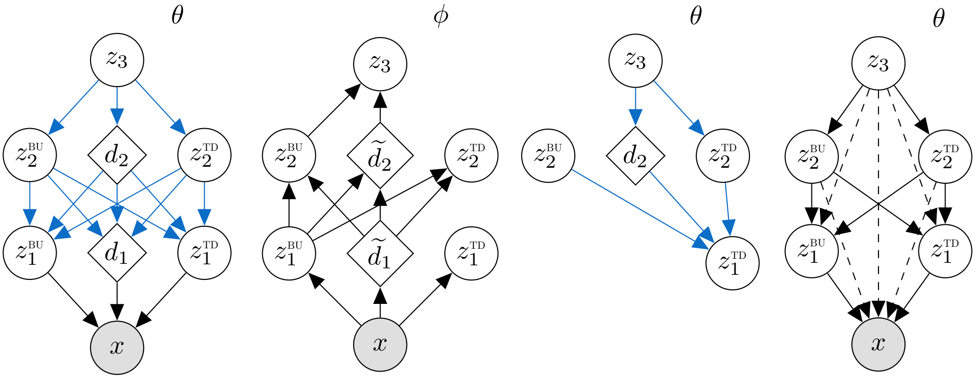


Figure 5.1: Generative

Figure 5.2: BU inference

Figure 5.3: TD inference

Figure 5.4: Dependencies

Figure 5.5: A $L = 3$ layered BIVA with **(a)** the generative model, **(b)** bottom-up (BU) inference path,**(c)** top-down (TD) inference path, and **(d)** variable dependency of the generative models where dashed lines denote a skip-connection. Blue arrows indicate that the deterministic parameters are shared within the generative model or between the generative and inference model. The diamonds represent the outputs of neural networks, the white circles indicate latent variables.

Autoregressive generative models parameterized by deep neural networks such as recurrent neural networks, convolutional neural networks or Transformers excel at modelling images (PixelCNN++, Salimans et al. 2017), text (XLNet, Yang et al. 2019) and audio (Sparse Transformer, Child et al. 2019). Nonetheless, controlling the generative process or inferring such attributes in autoregressive models is challenging, unless the training data is labelled with a set of known features (e.g., a class label describing the content of an image, or a conversation topic).

Generative models can be augmented with latent variables capturing global variation factors. Variational Autoencoders (VAE, Kingma and Welling 2014a) have proven to be well adapted to the image domain. This is especially true for the deep VAEs, which consist of stacking many layers of latent variables.[1] However, in the language domain, VAEs remain largely outperformed by LSTM or Transformer-based language models in generation and representation learning tasks.

In this chapter, we introduce BIVA: the deep hierarchical VAE introduced in Paper A. We include an overview of the architecture and discuss its application to language modelling. We refer to Paper A for an in-depth description of the method and a detailed review of the empirical results.

1 Architecture

BIVA (*Bidirectional-Inference Variational Autoencoder*) is a VAE which prior factorizes as a *top-down* deep hierarchy of latent variables. The inference network is decomposed into two parts: one that adopts a *bottom-up* factorization of the latent variables and another that adopts the reverse order: a *bottom-up* factorization. See Figure 5.5 for an illustration of the architecture.

Generative model The generative model of BIVA is identical to the one of a standard hierarchical VAE with skip connections. Given an observed variable $\mathbf{x}$ and a depth L, it is defined as:

$$p_\theta(\mathbf{x}, \mathbf{z}) := p_\theta(\mathbf{x} \mid \mathbf{z})\, p_\theta(\mathbf{z}_L) \prod_{i=1}^{L-1} p_\theta(\mathbf{z}_i \mid \mathbf{z}_{>i}) \,, \tag{5.1}$$

where each variable $\mathbf{z}_1, \ldots, \mathbf{z}_L$ corresponds to a level in the hierarchy. Each transition $p_\theta(\mathbf{z}_i \mid \mathbf{z}_{>i})$ is a Gaussian distribution parameterized by Gated ResNet blocks composed of convolutional neural networks (CNNs) [2] with skip-connections letting each layer of index i receive information from the above variables $\mathbf{z}_{>i} = \{\mathbf{z}_{i+1}, \ldots \mathbf{z}_L\}$. The top layer is a simple Gaussian distribution $p_\theta(\mathbf{z}_L) = \mathcal{N}(\mathbf{z}_L \mid 0, I)$.

Bidirectional inference network On the inference side of the model, each latent variable $\mathbf{z}_i$ in the hierarchy with position $i < L$ is decomposed into two parts, one for the *top-down* path and another for the *bottom-up* path:

$$\mathbf{z}_i = [\mathbf{z}_i^{\mathrm{TD}}; \mathbf{z}_i^{\mathrm{BU}}], \quad i \in [1, L-1] \,, \tag{5.2}$$

---

[1] Read more about deep VAEs in Child 2021; Kingma et al. 2016; Maaløe et al. 2019; Sønderby et al. 2016; Vahdat and Kautz 2020.

[2] The design of the Gated ResNet was inspired from Kingma et al. 2016

This factorization splits the hierarchy into two paths, which each factorize in opposite directions: a bottom-up path (i.e., $p(\mathbf{z}) = \prod_i p(\mathbf{z}_i \mid \mathbf{z}_{<i})$) and a top-down path (i.e. $p(\mathbf{z}) = \prod_i p(\mathbf{z}_i \mid \mathbf{z}_{>i})$). Both paths are parameterized by Gated ResNet blocks and the top-down path shares parameter with the generative model, as originally done in Sønderby et al. 2016. Combining both paths and conditioning the top-down on the bottom-up variables, we obtain a flexible bi-directional hierarchical posterior:

$$q_\phi(\mathbf{z} \mid \mathbf{x}) := q_\phi(\mathbf{z}_L \mid \mathbf{x}, \mathbf{z}^{\mathrm{BU}}_{<L}) \prod_{i=1}^{L-1} \underbrace{q_\phi(\mathbf{z}^{\mathrm{BU}}_i \mid \mathbf{x}, \mathbf{z}^{\mathrm{BU}}_{<i})}_{\text{bottom-up}} \underbrace{q_{\phi,\theta}(\mathbf{z}^{\mathrm{TD}}_i \mid \mathbf{x}, \mathbf{z}^{\mathrm{TD}}_{>i}, \mathbf{z}^{\mathrm{BU}}_{<i}, \mathbf{z}^{\mathrm{BU}}_{>i})}_{\text{top-down}} \ . \tag{5.3}$$

Optimization and evaluation In Paper A, all models are trained using the pathwise estimator (Section 2.1) applied to a modified evidence lower bound. The objective is modified with *freebits* (Kingma et al. 2016) to mitigate *posterior collapse*.[3]

2 Image modelling

In Table 5.1, we report the test likelihood of BIVA on CIFAR-10 with an updated list of baselines. Compared to the methods released prior to Paper A, BIVA performed comparably with the powerful but cumbersome autoregressive models (PixelCNN++). Looking at the methods published after Paper A, IAF-VAE and BIVA seem to have initiated a trend of deepening hierarchies of latent variables; deeper VAEs (NVAE: L=30, VD-VAE: L=45) were released soon after BIVA and were quickly followed by diffusion models (DPPM: L=1k, VDM: 10k). In Paper A, we report numerous additional image modelling experiments, including semi-supervised learning and anomaly detection.

3 Language modelling

In Paper A, we investigated whether BIVA could be applied to model 40-word sentences gathered from the BookCorpus dataset (Zhu et al. 2015). Text VAEs (Bowman et al. 2016; Semeniuta, Severyn, and Barth 2017) are designed with an LSTM language model as decoder. Such models have been notoriously difficult to optimize due to the decoder ignoring the latent variable, which is also known as the problem of *posterior collapse*.[4] Bowman et al. 2016 and Semeniuta, Severyn, and Barth 2017 bypassed this issue using strong regularization methods consisting of annealing the KL term in the ELBO (*deterministic warm-up*, Bowman et al. 2016; Sønderby et al. 2016) and weakening the decoder (*word dropout*).

---

[3] i.e. $\mathcal{D}_{\mathrm{KL}}(q_\phi(\mathbf{z} \mid \mathbf{x}) \,\|\, p_\theta(\mathbf{z}))$ collapsing to zero, see Section 1

[4] Read more about posterior collapse in Section 1.

	Type	**Depth**	**bits/dim**
With autoregressive components			
IAF-VAE Kingma et al. 2016	VAE	12	≤ 3.12
VLAE Chen et al. 2017b	VAE	1	≤ 2.95
PixelRNN Oord, Kalchbrenner, and Kavukcuoglu 2016	AR	–	$= 3.00$
PixelCNN++ Salimans et al. 2017	AR	–	$= 2.92$
PixelSNAIL† Chen et al. 2018	AR	–	$= 2.85$
Sparse Transformer† Child et al. 2019	AR	–	$= \mathbf{2.80}$
Without autoregressive components			
DiscreteVAE++ Vahdat et al. 2018	VAE	4	≤ 3.38
GLOW Kingma and Dhariwal 2018	Flow	–	$= 3.35$
Flow++ Ho et al. 2019	Flow	–	$= 3.08$
BIVA Maaløe et al. 2019	VAE	15	≤ 3.08
NVAE† Vahdat and Kautz 2020	VAE	30	≤ 2.91
Very Deep VAE† Child 2021	VAE	45	≤ 2.91
DDPM† Ho, Jain, and Abbeel 2020	Diff.	1k	≤ 3.70
VDM† Kingma, Salimans, Poole, et al. 2021	Diff.	10k	$\leq \mathbf{2.65}$

†Contemporary work or published after BIVA.

Table 5.1: Test log-likelihood on CIFAR-10 for BIVA and other deep generative models. The likelihood is approximate using the importance-weighted bound with $K = 1000$. We report the number of layers of latent variables when applicable (depth) as well as the type of model (autoregressive, VAE, flow, diffusion). We split the table between models that use autoregressive components (in the prior or in the decoder) and the others.

Latent variable language model BIVA is adapted to sequential data by replacing the 2D convolutions in the Gated ResNets with 1D convolutions. As done in related work, we augmented BIVA with an autoregressive conditional language model:

$$p_\theta(\mathbf{x} \mid \mathbf{z}) := \prod_{t=1}^{T} p_\theta(x_t \mid x_{<t}, \mathbf{z}) \ , \tag{5.4}$$

where $\mathbf{x}$ is a sentence encoded at the word level into q sequence $x_1, \ldots, x_T$ and $p_\theta(x_t \mid x_{<t}, \mathbf{z})$ is parameterized by an LSTM (Hochreiter and Schmidhuber 1997). We also experimented without LSTM decoder, in that case we define $p_\theta(\mathbf{x} \mid \mathbf{z}) := \prod_{t=1}^{T} p_\theta(x_t \mid \mathbf{z})$.

Key findings Table 5.2 reports the likelihood and latent variable usage measured by the KL divergence $\mathcal{D}_{\mathrm{KL}}(q_\phi(\mathbf{z} \mid \mathbf{x}) \| p_\theta(\mathbf{z}))$ of BIVA and the baseline models under comparable parameter budgets and using the same regularization technique (*freebits* + fine-tuning without regularization). In terms of likelihood performances, BIVA without autoregressive component is largely outperformed by the purely autoregressive LSTM language model. Coupled with an autoregressive component, BIVA and the other text VAEs bridge the gap with the LSTM baseline. BIVA performed comparably with the other text VAEs, although we report a higher usage of the latent variables for BIVA, as measured by the KL divergence.

	Parameters	$-\log p_\theta(\mathbf{x})$	**KL**
Results with autoregressive components			
LSTM	$15.0M$	$= \mathbf{41.49}$	$-$
LSTM-VAE	$23.7M$	≤ 42.41	5.13
Hybrid-VAE	$23.7M$	≤ 42.24	4.67
BIVA L=7	$23.0M$	≤ 42.34	**10.15**
Results without autoregressive components			
Hybrid-VAE	$15.0M$	≤ 54.53	14.10
BIVA L=7	$14.0M$	$\leq \mathbf{54.13}$	**15.33**

Table 5.2: Language modelling with BIVA. We report the test likelihood (measured with the ELBO) and the KL divergence $\mathcal{D}_{\mathrm{KL}}(q_\phi(\mathbf{z} \mid \mathbf{x}) \,\|\, p_\theta(\mathbf{z}))$. BIVA achieves higher KL divergences than the baseline models for a comparable likelihood, indicating a better conditioning of the decoder on the latent variables. LSTM-VAE is presented in Bowman et al. 2016. Hybrid VAE combines a CNN encoder with a CNN-LSTM decoder (Semeniuta, Severyn, and Barth 2017).

4 Conclusion

We presented BIVA; an example of a deep Variational Autoencoder. We discussed image modelling and text modelling experiments. BIVA excelled at modelling images, performing comparably with powerful autoregressive models. However, in the text domain, deep VAEs might not be sufficient, and it might not be possible to obtain good generative performances without autoregressive components. Hybrid VAEs, such as BIVA with an LSTM decoder, can learn to condition the text generative process on latent variables but the latent variable doesn't lead to improved likelihood performances.

Although the text experiment was limited in its design due to the relatively small scale of the model, the choice to impose a fixed-parameter budget, the use of a small dataset and the evaluation objective (ELBO), this introductory text experiment hints that latent variable models developed for the image models might not be so easily applied to text modelling.

In the years following Paper A, Transformers have become an indispensable component for text modelling. In the next chapter, we will present alternatives to the standard hybrid VAE architecture (e.g., Bowman et al. 2016; Maaløe et al. 2019; Semeniuta, Severyn, and Barth 2017) that allow harnessing the full power of Transformers for latent variable language modelling.

CHAPTER 6

Language as a Latent Variable

This chapter cites two of the contributions in this thesis:

Paper C: “Can large language models reason about medical questions?” (Liévin, Hother, and Winther 2022)

Paper D: “Variational Open-Domain Question Answering” (Liévin et al. 2022b)

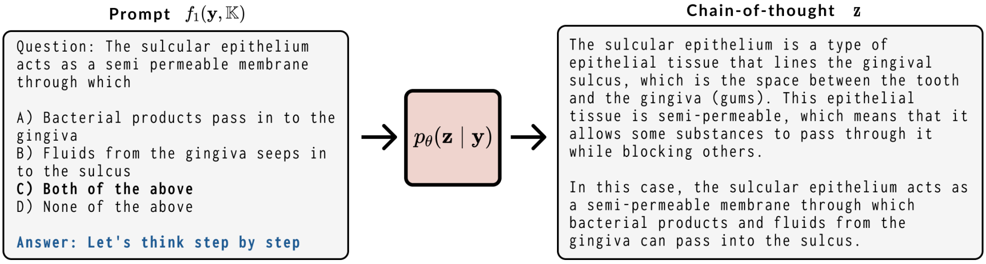
In this chapter, we present examples of latent variable models with structured and interpretable latent spaces. Although latent variables are unobserved, knowledge about the underlying generative process can be encoded in the design of the generative process. Miao and Blunsom 2016 is an excellent example of a stochastic generative model which relies on *language as a latent variable*. This generative process breaks down into two steps (i) generating keywords (latent variable) and (ii) generating a full-length text based on the keywords (observation).

In this thesis, we explored two types of models which latent variables are text data. In Paper C, the prior is a large language model that generates possible step-by-step solutions to a given written problem. In Paper D, the latent space is defined as a large collection of documents (e.g., Wikipedia). In the following sections, we present the architectures of the two models. Each method is discussed in depth in Chapters 10 and 9.

1 Latent Reasoning

This section gives an overview of the zero-shot reasoning language model from Paper C. The model and the topic of prompt-based learning and zero-shot reasoning are discussed in greater depth in Chapter 9 and Paper C.

Prompt-based learning Large language models such as GPT-3 (Brown et al. 2020a) are autoregressive left-to-right language models (i.e. $p_\theta(\mathbf{x}) = \prod_t p_\theta(x_t \mid x_{<t})$) consisting of a large number of parameters (175 billion parameters) and trained on massive text datasets. Prompt-based learning involves designing text instructions, dubbed *prompts*, that trigger pre-trained language models to generate predictions.


Prompt $f_1(\mathbf{y}, \mathbb{K})$
$p_\theta(\mathbf{z} \mid \mathbf{y})$
Chain-of-thought $\mathbf{z}$

```
Question: The sulcular epithelium
acts as a semi permeable membrane
through which

A) Bacterial products pass in to the
gingiva
B) Fluids from the gingiva seeps in
to the sulcus
C) Both of the above
D) None of the above

Answer: Let's think step by step
```

```
The sulcular epithelium is a type of
epithelial tissue that lines the gingival
sulcus, which is the space between the tooth
and the gingiva (gums). This epithelial
tissue is semi-permeable, which means that it
allows some substances to pass through it
while blocking others.

In this case, the sulcular epithelium acts as
a semi-permeable membrane through which
bacterial products and fluids from the
gingiva can pass into the sulcus.
```

Figure 6.1: Reasoning about a medical question using conditional language models $p_\theta(\mathbf{z} \mid \mathbf{y})$. The question is taken from the MedMCQA dataset, the chain-of-thought (reasoning) was generated using GPT-3. The text in blue is a text instruction appended to the question (i.e. zero-shot chain-of-thought prompt; Kojima et al. 2022).

Given $\mathbf{y}$ a task input (e.g., question) and $\mathbb{K}$ the variable representing the additional knowledge encoded into the prompt (e.g., task instructions, task examples or domain knowledge), we denote f the function that combines the variables $\mathbf{y}$ and $\mathbb{K}$ into a prompt denoted $f(\mathbf{y}, \mathbb{K})$, this is generally referred to as a *prompt template*.

Zero-shot reasoning *Zero-shot chain-of-thought* (CoT) prompting (Kojima et al. 2022) aims at generating a detailed step-by-step problem solutions, dubbed *chain-of-thoughts*. The method is conceptually simple and boils down to prompting a large language model with "*Let's think step by step*". See Figure 6.1 for an illustration.

Latent reasoning model Let's consider a question-answering setting with question $\mathbf{y}$ and answer $\mathbf{x}$ and an autoregressive language model p_θ. Denoting $\bar{\mathbf{y}} = f_1(\mathbf{y}, \mathbb{K})$ the zero-shot CoT prompt (Figure 6.1) augmenting the question $\mathbf{y}$, the prior is

$$p_\theta(\mathbf{z} \mid \bar{\mathbf{y}}) = \prod_{t=1}^{|\mathbf{z}|} p_\theta(z_t \mid [\bar{\mathbf{y}} \,;\, z_{<t}]) \tag{6.1}$$

where $[\cdot;\cdot]$ denotes the concatenation operator, and $\mathbf{z}$ is defined on the text domain. The decoding part of the generative model answers the question based on the question $\mathbf{y}$ and the latent chain-of-thought $\mathbf{z}$. It relies on an answer extraction prompt $f_2(\mathbf{u}, \mathbb{K})$ (e.g., "$< \mathbf{u} >$, *therefore, the answer is*"). Combining the latent reasoning prior with the answer extraction decoder, we obtain the zero-shot answering model is:

$$p_\theta\big(\mathbf{x}, \mathbf{z} \mid \mathbf{y}, \mathbb{K}\big) := \underbrace{p_\theta\big(\mathbf{x} \mid f_2([\bar{\mathbf{y}} \,;\, \mathbf{z}], \mathbb{K})\big)}_{\text{answering step}} \; \underbrace{p_\theta\big(\mathbf{z} \mid \bar{\mathbf{y}}\big),}_{\text{reasoning step}} \quad . \tag{6.2}$$

The resulting marginal answering model corresponds to exploring the many reasoning paths that are likely under the large language model, which we express here as:

$$p_\theta\left(\mathbf{x} \mid \mathbf{y}, \mathbb{K}\right) = \mathbb{E}_{p_\theta(\mathbf{z}|\bar{\mathbf{y}})}\Big[p_\theta\left(\mathbf{x} \mid f_2([\bar{\mathbf{y}} \,;\, \mathbf{z}], \mathbb{K})\right)\Big] \,. \tag{6.3}$$

2 Latent Information Retrieval

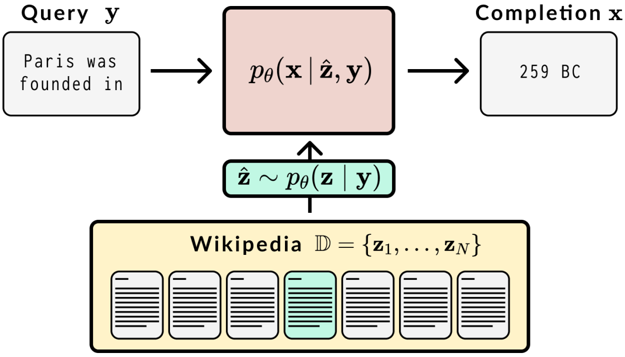

Query y
Paris was founded in
$p_\theta(\mathbf{x} \mid \hat{\mathbf{z}}, \mathbf{y})$
Completion x
259 BC
$\hat{\mathbf{z}} \sim p_\theta(\mathbf{z} \mid \mathbf{y})$
Wikipedia $\mathbb{D} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$

Figure 6.2: Augmenting language models with large knowledge bases.

Paper D experiments with the task of *open-domain question answering* (ODQA), in this section we detail the corresponding latent variable model and discuss its application to language modelling (Figure 6.2).[1]

Latent knowledge space Given a question $\mathbf{y}$ with answer $\mathbf{x}$ and a collection of N documents $\mathbb{D} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$, the ODQA task can be formulated as a latent variable model (Lee, Chang, and Toutanova 2019) with marginal likelihood defined as:[2]

$$p_\theta(\mathbf{x} \mid \mathbf{y}, \mathbb{D}) := \sum_{\mathbf{z} \in \mathbb{D}} p_\theta(\mathbf{x} \mid \mathbf{z},\, \mathbf{y})\, p_\theta(\mathbf{z} \mid \mathbf{y})\,, \tag{6.4}$$

where $p_\theta(\mathbf{x} \mid \mathbf{z}, \mathbf{y})$ *machine reading comprehension model* and where $p_\theta(\mathbf{z} \mid \mathbf{y})$ is a document retriever, both implemented using pre-trained language models.

Retrieval-augmented language modelling ODQA can be seen as a special case of latent variable masked language model. Given an input sequence $\mathbf{x}$ with masked tokens $\mathbf{x_m}$ and with remaining unmasked tokens $\mathbf{x_{-m}}$, a retrieval-augmented language modelling (REALM, Guu et al. 2020) is defined as:

$$p_\theta(\mathbf{x_m} \mid \mathbf{x_{-m}}) = \sum_{\mathbf{z} \in \mathbb{D}} p_\theta(\mathbf{x_m} \mid \mathbf{z},\, \mathbf{x_{-m}})\, p_\theta(\mathbf{z} \mid \mathbf{x_{-m}})\ , \tag{6.5}$$

which corresponds to the ODQA marginal likelihood (Equation 6.4) using $\mathbf{y} = \mathbf{x}_{-m}$ and $\mathbf{x} = \mathbf{x}_m$. Retrieval-augmentation can also be applied to autoregressive language

---

[1] Read more about open-domain question answering in Section 1.

[2] The notation of Paper D was adapted to make this chapter more consistent $(\mathbf{q} \mapsto \mathbf{y}, \mathbf{a} \mapsto \mathbf{x}, \mathbf{d} \mapsto \mathbf{z})$.

models, which was initially explored in Lewis et al. 2020 (RAG). This corresponds to retrieving one document $\mathbf{d}_t$ for each token x_t with a model:

$$p_\theta(\mathbf{x}) = \prod_{t=1}^{|\mathbf{x}|} \sum_{\mathbf{z}_t \in \mathbb{D}} p_\theta(x_t \mid \mathbf{z}_t,\, x_{<t})\, p_\theta(\mathbf{z}_t \mid x_{<t})\,. \tag{6.6}$$

3 Conclusion

We presented two examples of latent variable models with structured latent space. the latent spaces are defined in the text domain; on one hand, the text is generated, and on the other hand, the text is retrieved from a knowledge base. In both cases, the models are parameterized with pre-trained language models which serve specific functions: generating hypotheses, generating answers and retrieving information. This differs from the classical text VAEs presented in Chapter 5, in which the latent spaces are defined arbitrarily and mostly serve the purpose of defining a more flexible density function.

In open-domain question answering, the documents containing relevant knowledge are unknown. In the best scenario, optimizing the model via maximum likelihood leads to learning to read, comprehend and retrieve documents at scale. In the case of the latent reasoner, marginalizing the latent space leads to exploring many possible hypotheses. Thus optimizing structured latent variable models might be useful beyond building better language models; it can be applied to integrate and explore large knowledge spaces. In Chapter 10, we study the topic of ODQA in further detail and show that the learned retrievers can be applied to real-world information retrieval tasks.

The two models presented in this section rely on language as a latent variable. Text data is essentially discrete and therefore optimizing the corresponding latent variable models is challenging. This will be the theme of the next chapter, in which we introduce a new family of gradient estimators for discrete latent variable models.

Part III

Importance Weighted Bounds and Optimization

CHAPTER 7

Optimal Variance of the Score Function Estimator

This chapter cites one of the contributions in this thesis:

Paper B: Liévin et al. 2020 “Optimal Variance Control of the Score-Function Gradient Estimator for Importance-Weighted Bounds”

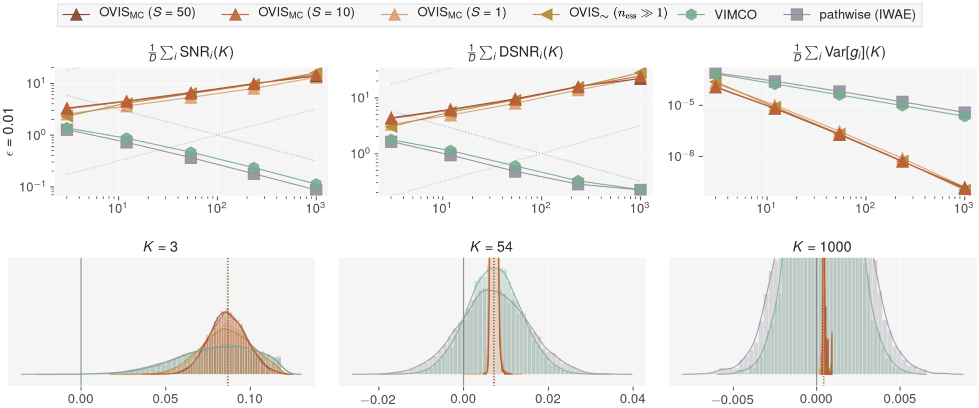


Figure 7.1: Signal-to-noise ratio (SNR), directional SNR (DSNR) and variance of gradients estimator applied to the importance-weighted bound for $K = 1 \ldots 1000$. We report the gradient statistics and visualise the distribution of gradient value for a random parameter, for three approximations of OVIS and two baselines: VIMCO and the pathwise gradient estimator. OVIS achieves lower variance than the two baseline estimators, resulting in an SNR that grows at a rate $\sqrt{K}$ whereas the SNR of the baselines decreases at a rate $1/\sqrt{K}$.

In this section, we study the variational inference for discrete latent variable models. This chapter begins with a background section, we motivate the use of unbiased estimators and discuss related methods. In Section 2, we introduce OVIS[1], a score function estimator with a more optimal baseline. We conclude this chapter with experimental results.

---

[1] OVIS: Optimal Variance – Importance Sampling

1 Background

Categorical distributions are not reparameterizable and therefore the low-variance pathwise gradient estimator (see Section 2.1) is not directly applicable. In Paper B, Appendix F, we review the gradient alternatives of gradient estimators for variational inference with discrete variables (VIMCO, RWS, REBAR, RELAX, TVO).

In this section, we first discuss some of the challenges intrinsic to training discrete latent variable models using continuous relaxations. Second, we introduce the VIMCO and RWS gradient estimators, which are directly related to our method; OVIS. Last we conclude the background section with a known defect of importance weighted bound estimators. We recommend expert readers to jump to Section 2 directly.

1.1 Continuous relaxations

Continuous relaxations of discrete variables such as the Gumbel-Softmax/Concrete distribution (Jang, Gu, and Poole 2017; Maddison, Mnih, and Teh 2017) allow approximating discrete distributions with a reparameterizable continuous one. Given a categorical distribution with probabilities $\Pi = [\pi_1, \ldots, \pi_N]$, the sampling process $\mathbf{z} \sim \text{Cat}(\Pi)$ is equivalent to the following process (Gumbel-Max trick):

$$\mathbf{z} = \text{one\_hot}\Big(\arg\max_i \left[g_i + \log \pi_i\right]\Big) \ , \qquad g_1, \ldots, g_N \sim \text{Gumbel}(0, 1) \tag{7.1}$$

The Concrete/Gumbel-Softmax relaxation consists in relaxing $\arg\max$ operator used in the Gumbel-Max sampling process. Relaxing Equation 7.1 yields a transformation of the Gumbel noise and the class probabilities into an N-dimensional continuous variable with each component $i \in [1, N]$ defined as:

$$\tilde{\mathbf{z}}_i = \frac{\exp\left(\left(\log \pi_i + g_i\right)/\tau\right)}{\sum_{i=1}^{N} \exp\left(\left(\log \pi_i + g_i\right)/\tau\right)} \ , \tag{7.2}$$

where $\tau > 0$ is a temperature parameter controlling the smoothness of the relaxation. This sampling process defines a valid continuous and reparametrizable probability distribution $p^\tau(\tilde{\mathbf{z}})$ (see Jang, Gu, and Poole 2017; Maddison, Mnih, and Teh 2017 for more details). Furthermore, the continuous distribution $p^\tau(\tilde{\mathbf{z}})$ converges to $\text{Cat}(\Pi)$ for $\tau \rightarrow 0$ and the continuous variable $\tilde{z}$ becomes sparser as τ is decreased, converging to a one-hot encoded value.

Continuous relaxations are effective for training relative simple deep generative models such as the VQ-VAE (Oord, Vinyals, and Kavukcuoglu 2017 with a flat prior). Read Sønderby, Poole, and Mnih 2017 for experimental results.

Relaxation bias In the workshop paper Liévin et al. 2022a, we explored training of more complex discrete VAEs. We found that problems are likely to arise when increasing the depth of the neural networks that parameterize the model. Using a relaxed approximate posterior $q^\tau_\phi(\tilde{\mathbf{z}} \mid \mathbf{x})$ with relaxed variable $\tilde{\mathbf{z}}$ and temperature $\tau > 0$ instead of a true categorical posterior $q_\phi(\mathbf{z} \mid \mathbf{x})$ results in evaluating the *relaxed* ELBO $\mathcal{L}^\tau(\mathbf{x}; \theta, \phi)$ instead of the ELBO $\mathcal{L}(\mathbf{x}; \theta, \phi)$ (see Section 2). The error induced by using the relaxed objective instead of the correct is quantified by the *relaxation bias*:

$$\delta^\tau(\mathbf{x}; \theta, \phi) := |\mathcal{L}^\tau(\mathbf{x}; \theta, \phi) - \mathcal{L}(\mathbf{x}; \theta, \phi)| \tag{7.3}$$

In Liévin et al. 2022a, we showed that, for a one layer VAE, the relaxation bias is upper bounded by a function of the Lipschitz constant $\kappa(\theta, \phi)$ of the function $\log p_\theta(\mathbf{x}, \mathbf{z}) - \log q_\phi(\mathbf{z} \mid \mathbf{x})$ and the expected difference between the relaxed variable $\tilde{\mathbf{z}}$ and its discretized counterpart $H(\tilde{\mathbf{z}})$:

$$\delta^\tau(\mathbf{x}; \theta, \phi) \leq \kappa(\theta, \phi)\ \mathbb{E}_{q^\tau_\phi(\tilde{\mathbf{z}}|\mathbf{x})}\left[\|\tilde{\mathbf{z}} - H(\tilde{\mathbf{z}}))\|_2\right] \ . \tag{7.4}$$

As a direct consequence of the above inequality is that evaluating discrete VAEs using continuous relaxations can results in an arbitrary large error unless the neural networks parameterizing the model are designed such that their Lipschitz constant remains small. This theoretical investigation, as well as the empirical observation that relaxed discrete latent variable models are difficult to train, motivate the research relaxation-free methods, discussed in the next sections.

1.2 Mutli-sample baselines for the score function estimator

we begin with recalling the importance-weighted bound (Section 3) and the score function estimator (Section 5) before introducing the VIMCO (Variational inference for Monte Carlo objectives) estimator.

Importance-weighted bound Given a generative model $p_\theta(\mathbf{x}, \mathbf{z})$, an approximate posterior $q_\phi(\mathbf{z} \mid \mathbf{x})$ and weights $w_{\theta,\phi}(\mathbf{x}, \mathbf{z}) := p_\theta(\mathbf{x}, \mathbf{z}) / q_\phi(\mathbf{z} \mid \mathbf{x})$, the importance-weighted bound (IWB) is defined for K samples as:

$$\mathcal{L}^K(\mathbf{x}) := \mathbb{E}_{q_\phi(\mathbf{z}_{1:K})}\left[\log \frac{1}{K} \sum_{i=1}^{K} w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_i)\right] \ . \tag{7.5}$$

Score function estimator with baseline In section 5, we derived estimators for both the parameter θ of the generative model (Equation 3.20 with $\alpha = 0$) and for the variational parameter ϕ. In this section, we focus on the score function gradient estimator, which corresponds to Equation 3.25 with $\alpha = 0$:

$$\bar{\eta}^{\text{reinforce}}_{\phi,K} := \sum_{i=1}^{K} \Big(d_i(\mathbf{x}, \mathbf{z}_{1:K}) - c(\mathbf{x}, \mathbf{z}_{-i})\Big) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x}), \quad \mathbf{z}_{1:K} \sim q_\phi(\mathbf{z} \mid \mathbf{x}) \tag{7.6}$$

where

$$d_i(\mathbf{x}, \mathbf{z}_{1:K}) := \log \hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K}) - \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i) \tag{7.7a}$$

$$\hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K}) := \frac{1}{K} \sum_{i=1}^{K} w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_i) \tag{7.7b}$$

The *baseline* c is chosen to minimize the variance of the gradient estimator and must satisfy $\mathbb{E}\left[c(\mathbf{x}, \mathbf{z}_{-i}) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x})\right] = 0$. See Section 5 for more details.

VIMCO Mnih and Rezende 2016 observed that the function $f_{\theta,\phi}(\mathbf{x}, \mathbf{z}_i)$ depends on two components, an unbounded term $\log \hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K})$ and a self-normalized term $\widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i) \in [0, 1]$. $\hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K})$ is a K-samples Monte Carlo estimate of the marginal likelihood denoted Z, which can also be estimated using the set of $K - 1$ samples $\mathbf{z}_{-i} \subset \mathbf{z}_{1:K}$.

The arithmetic version of the VIMCO estimator is a score function estimator (Equation7.6) with baseline defined as:

$$c(\mathbf{x}, \mathbf{z}_{-i}) := \log \hat{Z}(\mathbf{x} \mid \mathbf{x}_{-i}) = \log \frac{1}{K-1} \sum_{\substack{j=1 \\ j \neq i}}^{K} w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_j), \tag{7.8}$$

the other version (geometric average) uses a slightly different baseline (see the original paper).

VIMCO is computationally efficient because the values of the weights $w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_1)$, $\ldots$, $w_{\theta,\phi}(\mathbf{z}, \mathbf{z}_K)$ are computed one time and recycled to control each score component $f_{\theta,\phi}(\mathbf{x}, \mathbf{z}_i)$. VIMCO often deliver low variance gradient estimation because $c(\mathbf{x}, \mathbf{z}_{-i})$ is structurally similar to $f_{\theta,\phi}(\mathbf{x}, \mathbf{z}_i)$. Nonetheless, although the VIMCO baseline allows controlling the dominating component of the gradient, the other component remains uncontrolled. Indeed, the full gradient of VIMCO (arithmetic) is expressed as:

$$\bar{\eta}_{\phi,K}^{\text{VIMCO}} := \sum_{i=1}^{K} \Big(\underbrace{\log \hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K}) - \log \hat{Z}(\mathbf{x} \mid \mathbf{z}_{-i})}_{\text{controlled}} - \underbrace{\widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i)}_{\text{uncontrolled}} \Big) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x}) \ . \tag{7.9}$$

In Paper B, discussed in Section 2, we sough for the baseline that minimizes the variance of the score function estimator (Equation 7.6), without omitting the uncontrolled gradient component $-\sum_{i=1}^{K} \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x})$.

1.3 Reweighted Wake-Sleep

The Reweighted Wake-Sleep (RWS, Bornschein and Bengio 2015) is an importance-weighted re-interpretation of the original Wake-Sleep algorithm (Hinton et al. 1995) (See Paper B for details). The algorithm translates to optimizing the generative parameter θ and the variational parameter ϕ are optimized for different objectives. The parameter θ is optimized by differentiating and maximizing the importance-weighted bound $\mathcal{L}^K(\mathbf{x})$ (Equation 7.5). The parameter ϕ is optimized such as to maximize the KL divergence $\mathcal{D}_{\mathrm{KL}}\left(p_\theta(\mathbf{z} \mid \mathbf{x}) \,\|\, q_\phi(\mathbf{z} \mid \mathbf{x})\right)$ in the *wake* or *sleep* setting. Considering the *wake* phase only[2], the RWS gradient estimator for the variational parameter is defined as:[3]

$$\bar{\eta}^{\mathrm{RWS}}_{\phi,K} := \sum_{i=1}^{K} \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x}), \quad \mathbf{z}_{1:K} \sim q_\phi(\mathbf{z} \mid \mathbf{x}) \tag{7.10}$$

The RWS estimator has proven to be a powerful tool for variational inference with discrete latent variables and outperformed VIMCO on many tasks (Le et al. 2019).

1.4 Signal-to-noise ratio and importance-weighted bounds

Table 7.1: Asymptotic signal-to-noise ratio (SNR) of the unbiased importance-weighted gradient estimators. We indicate whether the estimator requires the variational distribution to be reparameterizable.

Parameter	**Estimator**	**Requires reparam.**	**SNR**
θ	Standard Monte Carlo	–	$\sqrt{K}$
ϕ	Pathwise Burda, Grosse, and Salakhutdinov 2016	✓	$1/\sqrt{K}$
ϕ	STL Roeder, Wu, and Duvenaud 2017	✓	$\sqrt{K}$
ϕ	DReG Tucker et al. 2019	✓	$\sqrt{K}$
ϕ	VIMCO Mnih and Rezende 2016	✗	$1/\sqrt{K}$
ϕ	OVIS Liévin et al. 2020	✗	$\sqrt{K}$

Rainforth et al. 2018 showed that, in the case of reparametrizable variational distributions, the pathwise gradient estimator of the importance weighted bound (Equation 3.22 with $\alpha = 0$) w.r.t. the parameter ϕ is essentially flawed. A study of the asymptotic signal-to-noise ratio (SNR) showed that, as the budget of Monte Carlo samples K increases, the pathwise estimator of the gradient w.r.t. θ improves whereas the gradient w.r.t. ϕ worsens. In Table, we report the asymptotic signal-to-noise ratio (SNR) for multiple estimators, including our estimator; OVIS, presented in the following section.

---

[2] Experiments from Le et al. 2019 suggest that the *wake* phase is generally more effective.

[3] To be consistent with the gradient of the ELBO, we report here the gradient of $-\mathcal{D}_{\mathrm{KL}}\left(p_\theta(\mathbf{z} \mid \mathbf{x}) \,\|\, q_\phi(\mathbf{z} \mid \mathbf{x})\right)$. Learning is assumed to be performed via stochastic gradient *ascent*.

2 OVIS

In this section, we will revisit the OVIS gradient estimators. We invite the readers to skip the presentation of the main theoretical results and instead use this updated version. Nonetheless, this section still refers to Paper B for the derivations.

The presentation of OVIS is decomposed as followed: (i) we show that a good asymptotic SNR is achievable for the score function estimator, (ii) we present an optimal but intractable optimal baseline for the score function estimator and (iii) we derive two classes of approximations of the optimal – but intractable – baseline.

2.1 Optimal asymptotic SNR

Similarly to Rainforth et al. 2018, in which the focus was on the pathwise estimator of the IW bound, we derived the asymptotic behavior of the score function estimator score ($\bar{\eta}_{\phi,K}^{\text{reinforce}}$, Equation 7.6) as $K \to \infty$ assuming an optimal baseline.

Decomposition of $d_i(\mathbf{x}, \mathbf{z}_{1:K})$ In Section 2.4, we discussed that a control variate $c(\mathbf{x}, \mathbf{z}_{-i})$ should not introduce dependencies on the sample $\mathbf{z}_i$ and therefore only depend on the set $\mathbf{z}_{-i}$. Thus, we introduce a decomposition of d_i into two parts:[4]

$$d_i(\mathbf{x}, \mathbf{z}_{1:K}) = f(\mathbf{x}, \mathbf{z}_i) + f(\mathbf{x}, \mathbf{z}_{-i}) \ . \tag{7.11}$$

Taylor expansion Following the methodology of Rainforth et al. 2018, assuming that the importance weights have finite variance, i.e. $\mathrm{Var}[w_{\theta,\phi}(\mathbf{x}, \mathbf{z})] < \infty$, we expand $\log \hat{Z}(\mathbf{z} \mid \mathbf{x}_{1:K})$ as a Taylor series of the variable $\hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K})$ around $Z(\mathbf{x}) := p_\theta(\mathbf{x})$.[5] By injecting the resulting Taylor expansion in $d_i(\mathbf{x}, \mathbf{z}_{1:K}) := \log \hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K}) - \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i)$, we can identify terms that depend on the weight $w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_i)$ and terms that depend on the other weights $\{w_{\theta,\phi}(\mathbf{z}, \mathbf{z}_j)\}_{j\neq i}$ and consequently express the terms $f(\mathbf{x}, \mathbf{z}_i)$ and $f(\mathbf{x}, \mathbf{z}_{-i})$ (Equation 7.11).

Asymptotic baseline For this analysis, we choose set the baseline to $c^{\infty}(\mathbf{x}, \mathbf{z}_{-i}) := f(\mathbf{x}, \mathbf{z}_{-i})$, that way we ensure the baseline to be independent of $\mathbf{z}_i$ (condition for the control variate to have zero expectation). Using the results of the Taylor expansion (Appendix B of Paper B), we get:

$$c^{\infty} := \log Z(\mathbf{x}) - \frac{3}{2} + \frac{2}{KZ(\mathbf{x})} \sum_{j\neq i} w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_j) - \frac{1}{2K^2 Z^2(\mathbf{x})} \left(\sum_{j\neq i} w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_j) \right)^2 \ . \tag{7.12}$$

---

[4] The functions $f(\mathbf{x}, \mathbf{z}_i)$ and $f(\mathbf{x}, \mathbf{z}_{-i})$ are unknown but assumed to exist.

[5] $\hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K}) \xrightarrow[K\to\infty]{} Z(\mathbf{x})$

Optimal SNR Although c^∞ is intractable because of its dependencies on the marginal likelihood $Z(\mathbf{x})$, the expression is sufficient to derive the optimal asymptotic signal-to-noise ratio. We define the optimal score function estimator (Equation 7.6) as the one using the optimal asymptotic baseline (Equation 7.12):

$$\bar{\mu}^{\infty}_{\phi,K} := \sum_{i=1}^{K} \Big(d_i(\mathbf{x}, \mathbf{z}_{1:K}) - c^{\infty}(\mathbf{x}, \mathbf{z}_{-i}) \Big) \nabla_\phi \log q_\phi(\mathbf{z}_i \mid \mathbf{x}) \ . \tag{7.13}$$

The variance, gradient and SNR of this estimator is (Appendix C of Paper B):

$$\mathbb{E}[\bar{\mu}^{\infty}_{\phi,K}] = \mathcal{O}(K^{-1}) \tag{7.14a}$$

$$\mathrm{Var}[\bar{\mu}^{\infty}_{\phi,K}] = \mathcal{O}(K^{-3}) \tag{7.14b}$$

$$\mathrm{SNR}[\bar{\mu}^{\infty}_{\phi,K}] = \mathcal{O}(\sqrt{K}) \ . \tag{7.14c}$$

Whereas Rainforth et al. 2018 predicted that the SNR of a pathwise estimator *decreases* at a rate of $1/\sqrt{K}$, our result shows that it is theoretically possible to obtain an SNR that *increases* at a rate of $\sqrt{K}$ using the score function estimator, and therefore without relying on reparameterizable variational distributions.

2.2 Optimal control variate

The baseline $c^\infty(\mathbf{x}, \mathbf{z}_{-i}) := f(\mathbf{x}, \mathbf{z}_{-i})$ was good enough to show that the score function estimator is not bound to worsen when K is increased. In this section, we report the result of the derivation of the optimal baseline, taking both parts $f(\mathbf{x}, \mathbf{z}_i)$ and $f(\mathbf{x}, \mathbf{z}_{-i})$ into account. In Paper B, Section 4, we derive the baseline $c_i^{\text{opt}}(\mathbf{x}, \mathbf{z}_{1:K})$[6] that minimizes the trace of the covariance of the score function estimator 7.13 based on the decomposition $d_i(\mathbf{x}, \mathbf{z}_{1:K}) = f(\mathbf{x}, \mathbf{z}_i) + f(\mathbf{x}, \mathbf{z}_{-i})$. At this step, we are not concerned about the bias of the estimator, were are interest in the analytical solution of the problem and let c_i^{opt} depend on all samples $\mathbf{z}_{1:K}$. Setting $\mathbf{h}(\mathbf{z}) = \nabla_\phi \log q_\phi(\mathbf{z} \mid \mathbf{x})$, we showed that:

$$c_i^{\text{opt}}(\mathbf{x}, \mathbf{z}_{1:K}) := \arg\min_{c_i} \operatorname{tr}\Big(\mathrm{Cov}\Big[\bar{\eta}^{\text{reinforce}}_{\phi,K}(c_i) \Big] \Big) \tag{7.15}$$

$$= \arg\min_{c_i} \operatorname{tr}\Big(\mathrm{Cov}\Big[\sum_{i=1}^{K} \big(f(\mathbf{x}, \mathbf{z}_i) + f(\mathbf{x}, \mathbf{z}_{-i}) - c_i(\mathbf{x}, \mathbf{z}_{1:K}) \big) \mathbf{h}(\mathbf{z}_i) \Big] \Big)$$

$$= f(\mathbf{x}, \mathbf{z}_{-i}) + \sum_{j=1}^{K} \frac{\mathbb{E}_{q_\phi(\mathbf{z}' \mid \mathbf{x})} \left[f(\mathbf{x}, \mathbf{z}_j)\, \mathbf{h}(\mathbf{z}')^T \mathbf{h}(\mathbf{z}_i) \right]}{\mathbb{E}_{q_\phi(\mathbf{z}' \mid \mathbf{x})} \left[\|\mathbf{h}(\mathbf{z}')\|^2 \right]} \ . \tag{7.16}$$

2.3 Approximations to the optimal control variate

Evaluating the optimal baseline $c_i^{\text{opt}}(\mathbf{x}, \mathbf{z}_{1:K})$ (Equation 7.16) requires marginalizing over $q_\phi(\mathbf{z} \mid \mathbf{x})$ and the corresponding control variate $\beta(\mathbf{x}, \mathbf{z}_{1:K}) =$

---

[6] We add the subscript i to keep track of the index of the gradient component being stabilized.

$\sum_i c_i^{\text{opt}}(\mathbf{x}, \mathbf{z}_{1:K})\mathbf{h}(\mathbf{z}_i)$ is not guaranteed to have zero expectation because $c_i^{\text{opt}}(\mathbf{x}, \mathbf{z}_{1:K})$ depends on $\mathbf{z}_i$. In this section, we introduce two classes of tractable approximations.

Monte Carlo simulation

In Paper B, Section 4, we show that given a parameter ϕ of large dimension, we can simplify Equation 7.16 to:[7]

$$\begin{aligned} c_i^{\text{opt}}(\mathbf{x}, \mathbf{z}_{1:K}) \approx & f(\mathbf{x}, \mathbf{z}_{-i}) + \mathbb{E}_{q_\phi(\mathbf{z}_i' \mid \mathbf{x})}\left[f(\mathbf{x}, \mathbf{z}_i')\right] && (7.17a) \\ = & \mathbb{E}_{q_\phi(\mathbf{z}_i' \mid \mathbf{x})}\left[f(\mathbf{x}, \mathbf{z}_{-i}) + f(\mathbf{x}, \mathbf{z}_i')\right] && (7.17b) \\ = & \mathbb{E}_{q_\phi(\mathbf{z}_i' \mid \mathbf{x})}\left[d_1(\mathbf{x}, [\mathbf{z}', \mathbf{z}_{-i}])\right] \ , && (7.17c) \end{aligned}$$

where the transition from the first line to the second is allowed because the variable $\mathbf{z}_i'$ is drawn independently of the set of samples $\mathbf{z}_{-i}$.

Using a set of S auxiliary samples allows us to define a Monte-Carlo estimate of the optimal baseline which resulting control variate has zero expectation:

$$c^{\text{MC}}(\mathbf{x}, \mathbf{z}_{-i}) := \frac{1}{S}\sum_{s=1}^{S} d_1(\mathbf{x}, [\mathbf{z}^{(s)}, \mathbf{z}_{-i}]) \quad \mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(S)} \sim q_\phi(\mathbf{z} \mid \mathbf{x}) \ . \tag{7.18}$$

High and low effective sample sizes

The effective sample size (ESS), denoted n_{eff} and introduced in Section 1.2, measures the quality of an importance sampling estimate. It roughly translates to the number of samples among $\mathbf{z}_1, \ldots, \mathbf{z}_K$ that are *effectively* used in the estimation of the target quantity. It is directly related to the entropy of the distribution of weights $\widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_1, \ldots, \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_K))$.

In paper B, Appendix D, we derived approximations to the optimal baseline in the ESS limits corresponding to $n_{\text{eff}} \gg 1$ and $n_{\text{eff}} \approx 1$. We report the results below.

High ESS In the limit $n_{\text{eff}} \gg 1$, a large number of weights have comparable values. Using a Taylor expansion of $\log \hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K})$ around $\frac{1}{K}\sum_{j \neq i} w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_j)$, the optimal baseline simplifies to:

$$c^{n_{\text{eff}} \gg 1}(\mathbf{x}, \mathbf{z}_{-i}) := \log \hat{Z}(\mathbf{x} \mid \mathbf{z}_{-i}) + \log(1 - \frac{1}{K}) \ . \tag{7.19}$$

For large values of K, the term $\log(1 - \frac{1}{K})$ simplifies to $-\frac{1}{K}$. Whereas the VIMCO baseline (Equation 7.8) controlled the term $\hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K})$ with an estimate $\hat{Z}(\mathbf{x} \mid \mathbf{z}_{-i})$ and left the term $-\widetilde{w_{\theta,\phi}}(\mathbf{z}, \mathbf{z}_i)$ uncontrolled, the baseline $c^{n_{\text{eff}} \gg 1}(\mathbf{x}, \mathbf{z}_{-i})$ extends VIMCO with an extra term $-\frac{1}{K}$. This term corresponds to the value of the weights

---

[7] The function $d(\mathbf{x}, [\mathbf{z}', \mathbf{z}_{-i}])$ outputs a vector of the same size as the input $[\mathbf{z}', \mathbf{z}_{-i}]$. Thus, $d_1(\mathbf{x}, [\mathbf{z}', \mathbf{z}_{-i}])$ corresponds to the output related to the first sample: $\mathbf{z}'$.

$-\widetilde{w_{\theta,\phi}}(\mathbf{z}, \mathbf{z}_i)$, which are roughly uniformly distributed in that setting. $c^{n_{\text{eff}} \gg 1}(\mathbf{x}, \mathbf{z}_{-i})$ is independent of $\mathbf{z}_i$ and therefore the resulting control variate has zero expectation; the gradient estimator is unbiased.

Low ESS In the limit $n_{\text{eff}} \approx 1$, only one weight dominate with a value of approximately one and the other weights are approximately zero. Let's denote $\star = \arg\max_{j \in [1,K]} w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_j)$ the index of the dominating weight. In this limit, the two terms of the prefactor $d_i(\mathbf{z}, \mathbf{z}_{1:K})$ can be approximated with:

$$\log \hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K}) \approx \log \frac{w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_\star)}{K} \qquad \text{and} \qquad \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i) \approx \mathbb{1}[i = \star] \ . \tag{7.20}$$

In Paper B, Appendix D, we show that in this ESS limit, the optimal baseline can be chosen as:

$$c_i^{n_{\text{eff}} \approx 1}(\mathbf{x}, \mathbf{z}_{1:K}) := \log \hat{Z}(\mathbf{x} \mid \mathbf{z}_{-i}) - \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i) \ . \tag{7.21}$$

The corresponding control variate has zero expectation when $n_{\text{eff}} = 1$ because

$$\mathbb{E}\left[\sum_{i=1}^{K} \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i) \mathbf{h}(\mathbf{z}_i)\right] = \mathbb{E}\left[\sum_{i=1}^{K} \mathbb{1}[i = \star] \mathbf{h}(\mathbf{z}_i)\right] = \mathbb{E}\left[\mathbf{h}(\mathbf{z}_\star)\right] = 0 \ .$$

when $n_{\text{eff}} > 1$, the control variate has a non-zero expectation and the resulting estimator is biased. In the next section, we investigate the nature of this bias.

2.4 OVIS for low ESS, RWS and STL

In the previous section, we have shown that for low ESS, using the optimal baseline ($c_i^{n_{\text{eff}} \approx 1}(\mathbf{x}, \mathbf{z}_{1:K})$ Equation 7.21) results in a biased score function estimator. The corresponding score function estimator can be decomposed using two of the gradient estimators introduced in Section 1:

$$\bar{\eta}_{\phi,K}^{n_{\text{eff}} \approx 1} = \underbrace{\sum_{i=1}^{K} \left(\log \hat{Z}(\mathbf{x}|\mathbf{z}_{1:K}) - \log \hat{Z}(\mathbf{x}|\mathbf{z}_{-i}) - \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i) \right) \mathbf{h}(\mathbf{z}_i)}_{\text{VIMCO (Equation 7.9)}} + \underbrace{\sum_{i=1}^{K} \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_i) \mathbf{h}(\mathbf{z}_i)}_{\text{RWS (Equation 7.10)}} \ . \tag{7.22}$$

Therefore, the bias of the OVIS estimator for low ESS corresponds to the gradient of the RWS objective, which itself corresponds to minimizing the divergence $\mathcal{D}_{\text{KL}}\left(p_\theta(\mathbf{z} \mid \mathbf{x}) \,\|\, q_\phi(\mathbf{z} \mid \mathbf{x})\right)$.

Interestingly, the bias of this gradient estimator is identical to the biased of the STL (Sticking The Landing, Roeder, Wu, and Duvenaud 2017) estimator.[8] Indeed, we have:

$$\bar{\eta}_{\phi,K}^{n_{\text{eff}} \approx 1} = \bar{\eta}_{\phi,K}^{\text{VIMCO}} + \bar{\eta}_{\phi,K}^{\text{RWS}} \tag{7.23a}$$

$$\bar{\eta}_{\phi,K}^{\text{STL}} = \bar{\eta}_{\phi,K}^{\text{pathwise}} + \bar{\eta}_{\phi,K}^{\text{RWS}} \tag{7.23b}$$

---

[8] We derive the STL estimator in Section 5

where $\bar{\eta}^{\text{pathwise}}_{\phi,K}$ is the pathwise gradient estimator applied to the importance weighted bound (Equation 3.22). Furthermore, because VIMCO and the pathwise estimators are unbiased, it follows that

$$\mathbb{E}\left[\bar{\eta}^{n_{\text{eff}}\approx 1}_{\phi,K}\right] = \mathbb{E}\left[\bar{\eta}^{\text{STL}}_{\phi,K}\right] \ . \tag{7.24}$$

Although the design OVIS for low ESS (Equation 7.21) is arbitrary for $n_{\text{eff}} > 1$, given that the baseline is only defined in $n_{\text{eff}} \to 1$, the link VIMCO, RWS and STL let us be more confident when applying this estimator to train deep generative models, where low ESS are often observed but remain superior to one.

3 Empirical Validation

We have studied the score function estimator and its optimal control variate theoretically. In Paper B, we confirmed our results empirically by replicating three landmark experiments of the literature. We showed that OVIS yields better SNR empirically, that OVIS learns better inference networks and that OVIS outperforms VIMCO.

3.1 Empirical SNR $\sim \sqrt{K}$

We replicated the small Gaussian model from Rainforth et al. 2018 that was utilized to showcase that "Tighter Variational Bounds are Not Necessarily Better." In Figure 7.1, we report the SNR of the unbiased OVIS estimators (MC and $n_{\text{eff}} \gg 1$) along with baselines and confirm our theoretical findings.

3.2 Gaussian mixture model

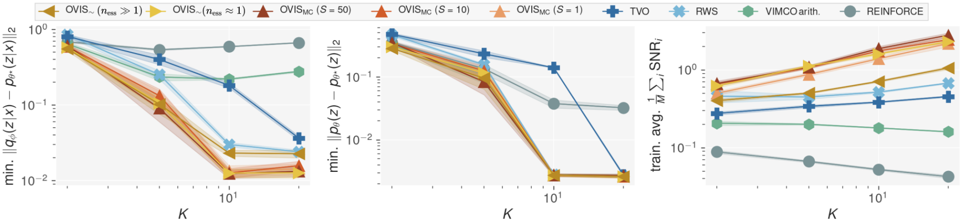


Figure 7.2: Training a simple Gaussian Mixture Model. OVIS estimators learn more optimal approximate posteriors (left) and yield higher SNR (right) than RWS, VIMCO and the TVO. The middle plot shows the difference between the learn prior parameters and the prior of the true generative process.

We replicated and trained the Gaussian model from Le et al. 2019 that was used to showcase that VIMCO learns poor approximate posteriors whereas RWS learns approximate posteriors that match the true posterior. Figure 7.2 shows that all OVIS estimators yield higher SNR and learn more optimal approximate posteriors (inference networks) than the RWS, VIMCO and TVO estimators.

3.3 Sigmoid Belief Network

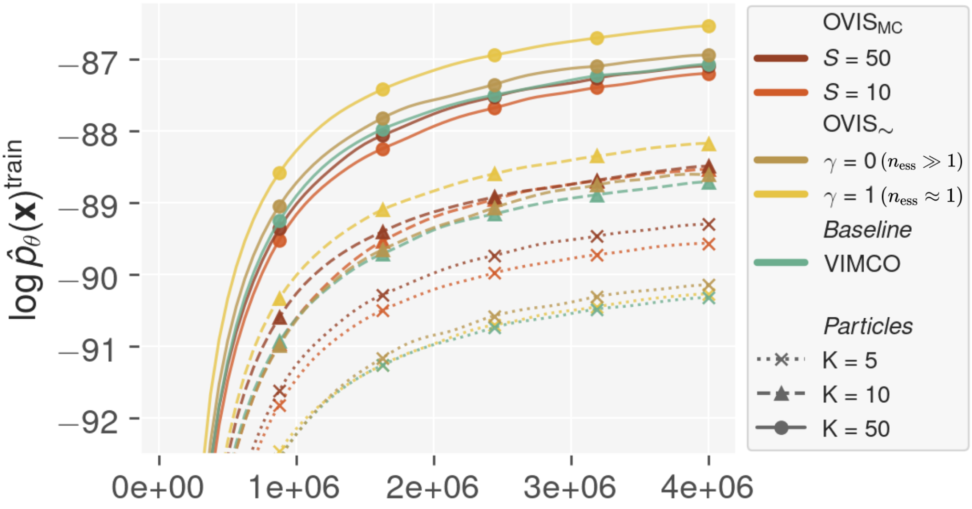


Figure 7.3: Training a Sigmoid Belief Network using OVIS (MC and ESS approximations) and VIMCO on Binarized MNIST (Salakhutdinov and Murray 2008). The marginal likelihood is approximated using 5000 Monte Carlo samples. OVIS approximation overall outperform VIMCO. The OVIS estimator for low ESS outperforms the other estimators for $K \geq 10$.

We replicated the Sigmoid Belief Network and experimental protocol from Mnih and Gregor 2014b; Mnih and Rezende 2016. In Figure 7.3, we report the training curves. We observed that all OVIS estimators outperform VIMCO, which confirms that using a more optimal control variate leads to better performances. Lastly, the OVIS estimator for low ESS outperformed VIMCO and the other OVIS estimator for $K \geq 10$.

4 Conclusion

OVIS is a novel control variate that provides unprecedented variance reduction of the score function estimator. The new variance reduction technique overcomes some of the flows of Monte Carlo estimators for the importance weighted bound, with a learning signal that increases as the number of Monte Carlo samples is increased and without requiring the variational distribution to be reparameterizable. Two classes of estimators were introduced, which should be chosen based on the application constraints (computational budget, ESS).

Our results allow us to recommend OVIS for the training of deep latent variable models such Sigmoid Belief Networks. In that setting, the computational budget is often limited and the Monte Carlo OVIS estimator cannot be applied. Instead, the class of sample-free approximations for low and high ESS will be used. Large and complex deep generative models are often found to suffer from a low ESS and therefore the OVIS estimator for low ESS (Equation 7.21) will be generally prefered.

Limitations In Paper B, we present additional experiments, including a comparison with the Thermodynamic Variational Objective (TVO, Masrani, Le, and Wood 2019) estimator. Although in this section, the empirical results were favourable to OVIS, OVIS estimators applied to the importance-weighted bound don't necessarily outperform the TVO.

When training deep generative models, the ESS is often measured close to one. Based on the definition of the ESS, this indicates that the importance sampling estimate $\hat{Z}(\mathbf{x} \mid \mathbf{z}_{1:K})$ has high variance.[9] In the limit $n_{\text{eff}} \approx 1$, given $\star$ the index of the dominating weight, OVIS for low ESS (Section 2.3) is approximately equal to

$$\bar{\eta}_{\phi,K}^{n_{\text{eff}}\approx 1} \approx \left(\log \frac{w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_\star)}{K} - \log \hat{Z}(\mathbf{x}|\mathbf{z}_{-\star}) \right) \mathbf{h}(\mathbf{z}_i) \tag{7.25}$$

where the difference $\log \frac{w_{\theta,\phi}(\mathbf{x},\mathbf{z}_\star)}{K} - \log \hat{Z}(\mathbf{x}|\mathbf{z}_{-\star})$ can be considerable and of arbitrarily large variance. Thus OVIS for low ESS would inevitably have high variance as well.

Nonetheless, this issue is not specific to the OVIS estimators, and is likely to apply to the importance-weighted bound in general. A low ESS means that the importance sampling estimate has a large variance, and therefore the corresponding gradient is prone to have a high variance as well. In Chapter 8, we show how to bypass such optimization challenges using Rényi divergence variational inference.

5 Bias of the STL Estimator for the Importance Weighted Bound

We consider a reparameterizable distribution $q_\phi(\mathbf{z} \mid \mathbf{x};)$ with noise distribution $p(\epsilon)$ and a sampling path $\mathbf{z}_\phi(\epsilon)$ such that $\mathbf{z} \sim q_\phi(\mathbf{z} \mid \mathbf{x};)$ is equivalent to $\mathbf{z}_\phi(\epsilon), \epsilon \sim p(\epsilon)$. Roeder, Wu, and Duvenaud 2017 observed that because $q(\mathbf{z}_\phi(\epsilon) \mid \mathbf{x}; \phi)$ is multivariate function that depending on ϕ two times, the gradients can be decomposed into the

---

[9] see the definition of the ESS in Section 1.2, Equation 2.10

path derivative and the *score function*:

$$\nabla_\phi = \log q_\phi(\mathbf{z}_\phi(\epsilon) \mid \mathbf{x}) = \underbrace{\frac{d\mathbf{z}_\phi(\epsilon)}{d\phi} \cdot \frac{\delta \log q_\phi(\mathbf{z} \mid \mathbf{x})}{\delta \mathbf{z}}\Big|_{\mathbf{z}=\mathbf{z}_\phi(\epsilon)}}_{\text{path derivative}} + \underbrace{\frac{\delta \log q_\phi(\mathbf{z} \mid \mathbf{x})}{\delta \phi}\Big|_{\mathbf{z}=\mathbf{z}_\phi(\epsilon)}}_{\text{score function}} . \tag{7.26}$$

Given $\epsilon_1, \ldots, \epsilon_K \sim q_\phi(\mathbf{z} \mid \mathbf{x})$ and setting $\tilde{w}_i = \widetilde{w_{\theta,\phi}}(\mathbf{x}, \mathbf{z}_\phi(\epsilon_i))$, by injecting Equation 7.26 in the gradient of the importance weighted bound, we obtain:

$$\begin{aligned}
\nabla_\phi \mathcal{L}^K(\mathbf{x}) &= \sum_i \tilde{w}_i \nabla_\phi \big[\log p_\theta(\mathbf{x} \mid \mathbf{z}_\phi(\epsilon_i)) + \log p_\theta(\mathbf{z}_\phi(\epsilon_i)) - \log q_\phi(\mathbf{z}_\phi(\epsilon_i) \mid \mathbf{x}) \big] \\
&= \sum_i \tilde{w}_i \nabla_\phi \big[\log p_\theta(\mathbf{z}_\phi(\epsilon_i) \mid \mathbf{x}) + \log p_\theta(\mathbf{x}) - \log q_\phi(\mathbf{z}_\phi(\epsilon_i) \mid \mathbf{x}) \big] \\
&= \sum_i \tilde{w}_i \nabla_\phi \big[\log p_\theta(\mathbf{z}_\phi(\epsilon_i) \mid \mathbf{x}) + \log p_\theta(\mathbf{x}) - \log q_\phi(\mathbf{z}_\phi(\epsilon_i) \mid \mathbf{x}) \big] \\
&= \overbrace{\sum_i \tilde{w}_i \nabla_{\mathbf{z}} \big[\log p_\theta(\mathbf{z}_\phi(\epsilon_i) \mid \mathbf{x}) - \log q_\phi(\mathbf{z}_\phi(\epsilon_i) \mid \mathbf{x}) \big] \cdot \nabla_\phi \mathbf{z}_\phi(\epsilon_i)}^{\text{path derivative}} \\
&\quad \underbrace{- \sum_i \tilde{w}_i \nabla_\phi \log q_\phi(\mathbf{z}_\phi(\epsilon_i) \mid \mathbf{x})}_{\text{score function}} \ .
\end{aligned}$$

STL consists in dropping the score function term; thus the bias is $\sum_i \tilde{w}_i \nabla_\phi \log q_\phi(\mathbf{z}_\phi(\epsilon_i) \mid \mathbf{x})$.

CHAPTER 8

Taming Importance Weighted Bounds using Rényi Divergences

This chapter cites two of the contributions in this thesis:

Paper B: "Optimal Variance Control of the Score-Function Gradient Estimator for Importance-Weighted Bounds" (Liévin et al. 2020)

Paper D: "Variational Open-Domain Question Answering" (Liévin et al. 2022b)

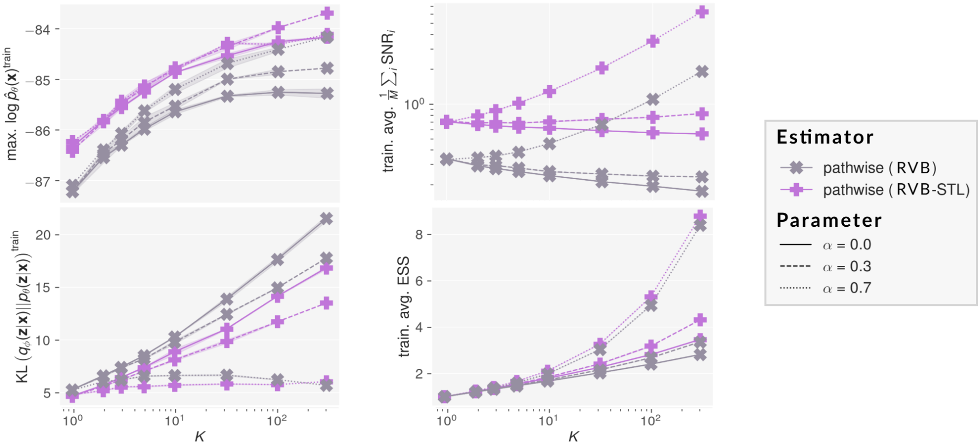


Figure 8.1: Training a one-layer Gaussian VAE with the pathwise and the STL estimators applied to the Rényi Variational Bound (RVB) for $\alpha \in \{0, 0.3, 0.7\}$ and different numbers of Monte Carlo samples $K = [1, 300]$. $\alpha > 1$ allows targeting higher ESS, reaching higher marginal likelihood and learning, better inference networks, as measured by the divergence $\mathcal{D}_{\text{KL}}(q_\phi \| p_\theta)$. We also report the SNR of the gradient, which grows with K for $\alpha = 0.3$. When α is set to zero, the RVB corresponds to the importance-weighted bound (IWB). For $\alpha = 0$ (RVB=IWB), the SNR decreases with K, even when using the STL estimator.

The importance-weighted bound (IWB) approximates the marginal likelihood more tightly as the number of Monte Carlo samples K is increased (Burda, Grosse, and Salakhutdinov 2016). At evaluation time, it is therefore common practice to evaluate as many samples as possible. However, during training, increasing the number of samples is not always beneficial, as it results in a worse learning signal (Rainforth et al. 2018).

In Chapter 7, we showed that, in the case of a system with a high effective sample size (ESS), one can improve the quality of the learning using OVIS control variates. However, when the ESS is close to one, even when using the OVIS control variates, maximizing the importance weighted bound might not lead to optimal learning. In the conclusion of Chapter 7), we discussed that this limitation is intrinsic to the importance weighted bound and therefore we sought tools that overcome this problem.

In this chapter, we argue that the Rényi variational bound (RVB), presented in Section 4, can be utilized to tame importance-weighted VAEs. Based on results from Papers B and D, we present empirical evidence supporting that the RVB can be applied to target higher ESS, which in turn yields higher SNR and results in superior learning performances.

1 Navigating Between Variational Bounds

Although the importance-weighted bound (IWB) provides a tighter marginal likelihood estimate than the evidence lower bound (ELBO), a tighter bound is not necessarily a better tool for optimization. In this section, we discuss two approaches to combining variational bounds.

1.1 *Inner* and *outer* samples

Rainforth et al. 2018 suggested combining variational bounds with linear combinations of bounds, or alternatively by evaluating the IWB using more than one *outer* samples. Given a number of K *inner* samples and M *outer* samples, the K-Samples IWB can be estimated using KM samples using the following estimate:

$$\hat{\mathcal{L}}^{M,K}(\mathbf{x}) := \frac{1}{M}\sum_{i=1}^{M} \log \frac{1}{K}\sum_{j=1}^{K} w_{\theta,\phi}(\mathbf{x}, \mathbf{z}_j^{(i)}), \quad \mathbf{z}_{1:K}^{(1)}, \ldots, \mathbf{z}_{1:K}^{(M)} \sim q_\phi(\mathbf{z} \mid \mathbf{x}) \ . \tag{8.1}$$

The pathwise gradient of the estimate $\hat{\mathcal{L}}^{M,K}(\mathbf{x})$ benefits from an inference network SNR in $\mathcal{O}(\sqrt{M/K})$. Therefore, under a particle budget MK, the bound $\hat{\mathcal{L}}^{M,K}(\mathbf{x})$ allows trading tightness of the bound for better SNR, which was confirmed by the authors empirically.

1.2 Rényi variational bound

Connecting bounds Similarly to the work of Rainforth et al. 2018, we proposed in Paper B to solve optimization issues using multiple variational objectives. For $\alpha \neq 1$, the RVB is defined as

$$\mathcal{L}_\alpha(\mathbf{x}) := \frac{1}{1-\alpha} \log \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, \mathbf{z}) \right] \ . \tag{8.2}$$

The VRB is continuous in α and is extended by continuity in $\alpha = 1$. In practice, the RVB is estimated with the importance-weighted Rényi bound (IWRB):

$$\mathcal{L}_\alpha^K(\mathbf{x}) := \frac{1}{1-\alpha} \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})} \left[\log \frac{1}{K} \sum_{i=1}^{K} w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, \mathbf{z}_i) \right] \tag{8.3}$$

which connects the ELBO, IWB and the marginal likelihood through the following equations:

$$\mathcal{L}_{\alpha \geq 0}^K(\mathbf{x}) \leq \log p_\theta(\mathbf{x}) \tag{8.4a}$$

$$\mathcal{L}_{\alpha=0}^K(\mathbf{x}) = \mathcal{L}^K(\mathbf{x}) \tag{8.4b}$$

$$\mathcal{L}_{\alpha=1}^K(\mathbf{x}) = \mathcal{L}(\mathbf{x}) \ . \tag{8.4c}$$

Rényi divergence variational inference offers a probabilistic framework to navigate between multiple variational bounds. Similarly to Rainforth et al. 2018, we argue that utilizing the ELBO, especially during early training, can benefit learning.

Effective sample size The expression of the RVB 8.2 provides additional insights as to why using looser bounds might be beneficial. The RVB relies on an importance sampling estimate $\mathbb{E}\left[w_{\theta,\phi}^{1-\alpha}(\mathbf{x}, \mathbf{z}) \right]$. When the parameter α is increased from zero to one, the distribution of weights is relaxed, and the weights become approximately equal, converging to $w_{\theta,\phi}^{0}(\mathbf{x}, \mathbf{z}) = 1$. Consequently, the effective sample size increases and the variance diminishes.

Guided training In $\alpha = 0$ the VRB aligns with the marginal likelihood. The value, and therefore the gradient, of VRB is independent of the approximate posterior $q_\phi(\mathbf{z} \mid \mathbf{x})$. However, in $\alpha = 1$ one, the VRB aligns with the ELBO, which value depends on the approximate posterior. When a good approximate posterior can be designed using domain knowledge, the ELBO can be utilized to constrain the optimization of the generative model. Indeed, considering the variational parameter known and fixed, maximizing the ELBO consists in maximizing the joint probability $p_\theta(\mathbf{x}, \mathbf{z})$ under $q_\phi(\mathbf{z} \mid \mathbf{x})$:

$$\nabla_\theta \mathcal{L}_{\alpha=1}(\mathbf{x}) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})} \left[\nabla_\theta \log p_\theta(\mathbf{x}, \mathbf{z}) \right] \ . \tag{8.5}$$

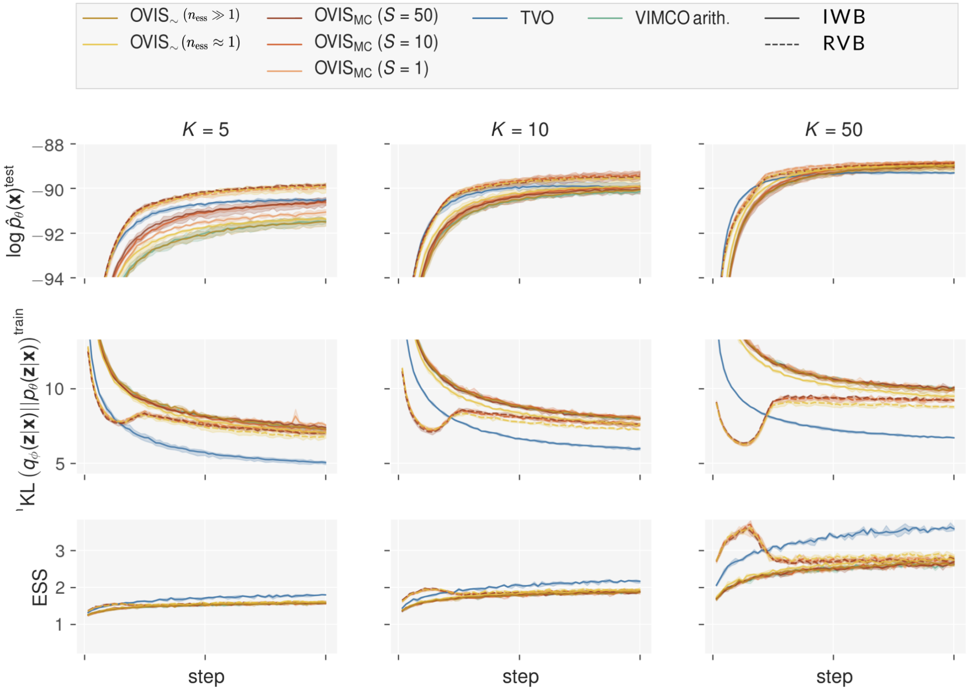


Figure 8.2: Training a Sigmoid Belief Network using OVIS, VIMCO and the TVO. We apply OVIS to the importance-weighted bound (IWB) and to the Rényi Variational Bound (RVB). The RVB allows targeting higher ESS (during initial training), higher marginal likelihood, and leads to learning better approximate posteriors, as measured by the KL divergence $\mathcal{D}_{\text{KL}}(q_\phi \| p_\theta)$.

2 Empirical results

Training deep latent variable models is challenging and the initialization often affects the whole course of the experiment. A low ESS might impair initial learning and the model might not recover with further training. In the section below, we compare the use of the ELBO, the IWB and the RVB for training VAEs.

2.1 Sigmoid Belief Network

In Paper B, we applied OVIS to the RVB and experimented with interpolating the parameter α from 0.99 to 0 during early training. We report the training dynamics in Figure 8.2. The RVB warm-up scheme appeared to be effective, as the models trained with RVB warm-up consistently reached higher marginal likelihood than the other models, including those trained using the TVO.

2.2 Gaussian VAE

We applied the pathwise estimator and STL (Roeder, Wu, and Duvenaud 2017) to the RVB and trained a one-layer Gaussian VAE (the experimental setup is reported in Paper B). In figure 8.1, we report the training curves for multiple values $\alpha = \{0, 0.3, 0.7\}$ and multiple number of samples $K = 1, 300]$. For both the estimators, the RVB with $\alpha = 0.7$ provided the best performances across the four considered metrics (marginal likelihood, $\mathcal{D}_{\mathrm{KL}}(q_\phi \| p_\theta)$, ESS and SNR).

2.3 Open-domain Question Answering

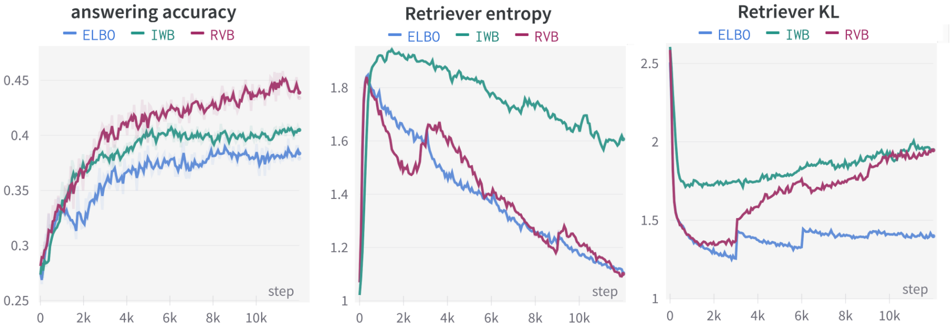


Figure 8.3: Training an open-domain question answering (ODQA) model using the evidence lower bound (ELBO), the importance weighted bound (IWB) and the Rényi variational bound (RVB). When using the RVB, the parameter α is decreased from one (ELBO) to zero (IWB) during the first 3k steps. We report, over training, the validation answering accuracy, the entropy of the retriever distribution and the KL divergence $\mathcal{D}_{\mathrm{KL}}(q_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q}) \| p_\theta(\mathbf{d} \mid \mathbf{q}))$. The RVB allows for reaching higher answering accuracy. During early training, using the RVB with $\alpha > 0$ constrains the retriever $p_\theta(\mathbf{d} \mid \mathbf{q})$ to match the approximate posterior $q_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$, resulting in smaller KL divergence between the two distributions.

In Paper D, we applied Rényi divergence variational inference to open-domain question answering (ODQA). In this experiment, the model is a VAE which prior $p_\theta(\mathbf{z})$ is a document retriever and which decoder $p_\theta(\mathbf{x} \mid \mathbf{z})$ is a machine reading comprehension model. The approximate posterior is a checkpoint of the retriever, updated every 3k steps, except for the first period of 3k steps where it corresponds to a domain-specific retriever model (BM25) used to guide the retriever.[1]

We experimented with (i) using the RVB and interpolating α from 1 to 0 during the first 3k steps, (ii) using the IWB and (iii) using the ELBO. In Figure 8.3, we report the performances and retriever characteristics observed during training. Using

---

[1] learn more about the implementation details in Chapter 10

the RVB affects the learning dynamics positively when compared to using the IWB and the ELBO (more details in the caption).

3 Conclusion

The RVB framework offers a theoretically grounded method for navigating between variational bounds. Although the idea of combining multiple bounds was introduced in Rainforth et al. 2018, using the RVB is more flexible and often more practical than combining inner and outer samplers.

We found the RVB to be a versatile tool that can be applied in various settings. In Papers B and D, we applied the RVB to training deep latent variable models successfully. In Paper 7, the RVB allowed for accelerating the learning of inference networks, leading to overall better generative models. In Paper D, optimizing with a looser bound allowed guiding the initial learning steps of the document retriever based on a reference domain-specific retriever model. Overall, our experiments showed that interpolating between variational bounds is effective at stabilizing the training of deep latent variable models and often lead to better performances.

Part IV

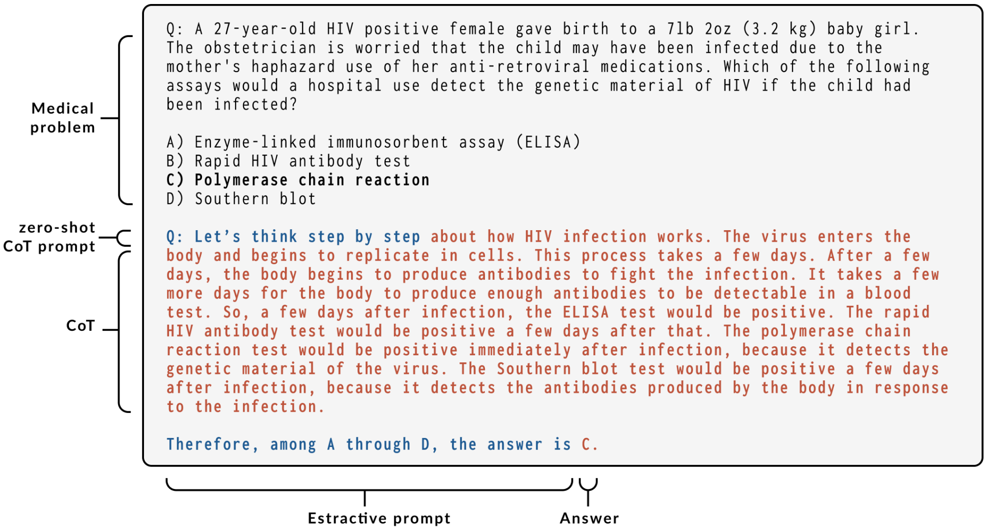
Medical Question Answering and Information Retrieval

CHAPTER 9

Zero-Shot Medical Reasoning using Large Language Models

This chapter cites one of the contributions in this thesis:

Paper C: “Can large language models reason about medical questions?” (Liévin, Hother, and Winther 2022)



Q: A 27-year-old HIV positive female gave birth to a 7lb 2oz (3.2 kg) baby girl. The obstetrician is worried that the child may have been infected due to the mother's haphazard use of her anti-retroviral medications. Which of the following assays would a hospital use detect the genetic material of HIV if the child had been infected?

A) Enzyme-linked immunosorbent assay (ELISA)
B) Rapid HIV antibody test
C) Polymerase chain reaction
D) Southern blot

Q: Let’s think step by step about how HIV infection works. The virus enters the body and begins to replicate in cells. This process takes a few days. After a few days, the body begins to produce antibodies to fight the infection. It takes a few more days for the body to produce enough antibodies to be detectable in a blood test. So, a few days after infection, the ELISA test would be positive. The rapid HIV antibody test would be positive a few days after that. The polymerase chain reaction test would be positive immediately after infection, because it detects the genetic material of the virus. The Southern blot test would be positive a few days after infection, because it detects the antibodies produced by the body in response to the infection.

Therefore, among A through D, the answer is C.

Figure 9.1: Answering a USMLE question with GPT-3 using zero-shot chain-of-thought (CoT) prompting (Kojima et al. 2022). The technique consists in prompting the language model two times: first to generate a CoT, second to generate an answer based on the question and the generated CoT. In this example, the CoT (*reasoning*) and the answer, in **red**, was generated by GPT-3. The question is displayed in **black** and the zero-shot CoT cues (instructions) are printed in **blue**.

Large language models (LLMs) such as GPT-3 Brown et al. 2020b, PaLM (Chowdhery et al. 2022) or Gopher (Rae et al. 2021) have acquired unprecedented natural language understanding capabilities. LLMs have achieved human-like performances on many NLP benchmarks, leading to many of them being considered obsolete (Srivastava et al. 2022). As language models are scaled to using billions of of parameters, language models acquire new abilities (e.g., arithmetic, natural language understanding or code generation) which were not observed in smaller language models (Ganguli et al. 2022). In Paper C, we investigated whether the advanced natural language understanding capabilities of large language models could be applied to solve challenging medical problems. In this chapter, we introduce our methods and review the main empirical findings.

1 Background

We begin with reviewing background information about prompt-based learning, zero-shot reasoning, retrieval augmentation and, last, introduce the two medical question-answering datasets that will be studied in this chapter. We recommend readers who are familiar with prompt-based learning to skip this section and move to Section 2 directly.

1.1 Prompt-based learning

Traditional language models such as BERT (Devlin et al. 2019) are pre-trained on large unstructured text datasets before being *fine-tuned* on task-specific labelled data. Thanks to unparalleled language proficiency, large language models can be applied to new tasks without the need for expensive task-specific fine-tuning.

The new learning paradigm; *pre-train, prompt and predict* (Liu et al. 2021), consists in crafting text instructions which completion will correspond to a prediction. The text instructions or *prompts* corresponds to a description of the task, which can be augmented using external knowledge and task examples (See Table 9.1). When incorporating task examples into the prompt, this corresponds to a *few-shot* learning setting, otherwise, it is referred to as *zero-shot* learning.

1.2 chain-of-thoughts

Prompts are not limited to encoding domain knowledge and can be engineered to instruct the model *how* to solve the task. In problems that require reasoning about the task and the available data (e.g., arithmetic, question answering), Wei et al. 2022 showed that it was possible to guide LLMs to break the problems into multiple reasoning steps via task-specific prompts. It translates to making the inner reasoning of an LLMs explicit, hence the term “chain-of-thought” (CoT).

Table 9.1: Examples of zero and few-shot prompt-based learning along with completions. Prompts can be engineered to augment the task with additional knowledge, instructions or examples. The task (question) is displayed in `black` , the cues augmenting the prompt are printed in `blue` and the completions are showed in `red`.

Task	**Prompt and completion**
QA	`Document: The Parisii settled on the banks of the Seine around 259 BC. Question: When was Paris founded? Answer: around 259 BC`
Translation	`French: Mon cheval est très rapide. English: My horse is very fast.`
Arithmetic	`2 + 3 = 5, 6 - 7 = -1, -3 + 5 = 2`

Although Wei et al. 2022 required task-specific prompts, Kojima et al. 2022 demonstrated that CoTs could be obtained using a single and domain-agnostic prompt "Let's think step by step". The technique, coined *zero-shot CoT*, is illustrated in Figure 9.1.

1.3 Grounding

Large language models memorise part of the knowledge embedded into the training data but might lack – or fail to recall – domain knowledge at inference time or *hallucinate* facts (Maynez et al. 2020). In Paper C, we experimented with grounding questions in the knowledge provided by factual documents. This corresponds to an instance of open-domain question answering, discussed in depth in Chapter 10.

1.4 Medical question answering datasets

Paper C applies prompt-based learning to three medical multiple-choice question-answering datasets. We provide below a brief introduction to the USMLE and the MedMCQA datasets. Read more about the question-answering datasets in Paper C.

USMLE The USMLE (MedQA) dataset Jin et al. 2021 gathers historical questions from the United States Medical Licensing Examination (USMLE), the examinations that medical professionals are required to pass before practising in the US.[1] The questions are notorious for being challenging as they often require strong problem-solving skills coupled with comprehensive medical knowledge. Each question features a description of a medical case and a question that emulates the real clinical setting.

MedMCQA The MedMCQA (Pal, Umapathi, and Sankarasubbu 2022) is a large-scale multiple-choice question answering collected from Indian medical school entrance

---

[1] `https://www.usmle.org/about-usmle`

exams (AIIMS and NEET-PG). The MedMCQA covers a broad range of medical topics (e.g., dentistry, psychiatry, surgery) and includes various problem types (e.g., logical, knowledge). The USMLE questions are often more difficult to answer than the MedMCQA questions, which tend to focus on evaluating core medical knowledge.

2 Medical Reasoning with Large Language Models

Paper C applies a number of prompt-based learning strategies to medical problems. In particular, we investigated few-shot learning (Brown et al. 2020a), zero-shot CoT (Kojima et al. 2022), retrieval augmentation (Lazaridou et al. 2022) and ensemble models (Wang et al. 2022). In this section, we present an overview of our method. Please refer to Paper C for further details.

2.1 Prompt engineering

We investigated the design of prompts along two main questions: (I) Can we adapt zero-shot CoT to the medical domain? and (II) How can we incorporate question-specific knowledge such as Wikipedia articles, examples of questions and CoTs into a single prompt?

CoT prompt In Paper C, we selected 30 variations of the original CoT prompt "*Let's think step by step*" such as "*Let's derive the differential diagnosis*" or " *Let's follow a Bayesian step by step approach*". Based on a small subset of USMLE questions, we benchmarked the 30 prompts and selected 5 of them, which are reported in the paper.[2]

Prompt templates In Table 9.2, we illustrate the templates utilized to build prompts based on questions, additional context (articles) and reference question-explanation-answer triplets (few-shot learning). We represent the zero-shot CoT setting as well as the CoT-free setting (denoted *zero-shot* $\emptyset$). The templates are a direct application of Brown et al. 2020a; Kojima et al. 2022; Lazaridou et al. 2022.

2.2 Generative models

In this section, we describe the model test in Paper C using the mathematical language introduced in Chapter 5, Section 1.

---

[2] Although more explicit prompts like " *Let's follow a Bayesian step by step approach*" didn't give substantially better performances, in some cases, this actually triggered GPT-3 to apply Bayes' rule (see Paper C, Table 12).

Table 9.2: Prompt templates utilized for multiple-choice question answering. In the table below, we display the `[provided data]` such as the question, additional context, or the answer in **black**, the `<completions>` generated by GPT-3 are printed **red**. We use `OR` to indicate when a variable can be either `[provided data] OR` a `<completions>`. The prompt `cues` (instructions) are displayed in **blue**. The symbol $\emptyset$ is used to represent empty strings.

	Zero-shot $\emptyset$	**Zero-shot CoT**
Question	`Question: [Question]`	`Question: [Question]`
CoT	$\emptyset$	`Answer: Let's think step by step <CoT>`
Answer	`Answer: among A through D, the answer is <answer>`	`Therefore, among A through D, the answer is <answer>`
	Zero-shot $\emptyset$ + grounding	**One-shot CoT**

Shot	$\emptyset$
	$\emptyset$
	$\emptyset$
		`Question: [Question]`
		`Answer: Let's think step by step [Explanation] OR <CoT>`
		`Therefore, among A through D, the answer is [answer]`

Context	`Context: [Context]`	$\emptyset$
Question	`Question: [Question]`	`Question: [Question]`
CoT	$\emptyset$	`Answer: Let's think step by step <CoT>`
Answer	`Answer: among A through D, the answer is <answer>`	`Therefore, among A through D, the answer is <answer>`

Single-answer likelihood Reusing the notation from Chapter 5, we denote $\mathbf{y}$ a question, $\mathbf{x}$ its answer and $\mathbf{z}$ a CoT. Given $\mathbb{K}$ a variable representing all the task augmentations (e.g., task instructions, task examples, Wikipedia articles), we denote $f_1(\mathbf{y}, \mathbb{K})$ the CoT prompt and $f_2(\mathbf{u}, \mathbb{K}))$ the answer extraction prompt (see Figure 9.1). f_1 and f_2 represent the template functions that allow composing the task augmentations $\mathbb{K}$ with the text input into a prompt. Using this notation, we express the generative reasoning model as:[3]

$$p_\theta\big(\mathbf{x}, \mathbf{z} \mid \mathbf{y}, \mathbb{K}\big) := \underbrace{p_\theta\big(\mathbf{x} \mid f_2([\bar{\mathbf{y}} \,;\, \mathbf{z}], \mathbb{K})\big)}_{\text{answering step}} \; \underbrace{p_\theta\big(\mathbf{z} \mid \bar{\mathbf{y}}\big)}_{\text{reasoning step}} \;, \quad \bar{\mathbf{y}} := f_1(\mathbf{y}, \mathbb{K}) \;. \tag{9.1}$$

Ensemble model In the continuity of Wang et al. 2022, we explored combining multiple prompts into an ensemble model. This corresponds to using P different task augmentations $\mathbb{K}_{1:P} = \{\mathbb{K}_1, \ldots, \mathbb{K}_P\}$ (e.g., choice of CoT prompt, additional Wikipedia passages). Using weights $\alpha_1 = \ldots = \alpha_P = 1/P$, we define the ensemble model as:

$$p_\theta(\mathbf{x} \mid \mathbf{y}, \mathbb{K}_{1:P}) := \sum_{i=1}^{P} \alpha_i \, p_\theta(\mathbf{x} \mid \mathbf{y}, \mathbb{K}_i) \;. \tag{9.2}$$

3 Main Results

In Paper C, we experimented with zero-shot, few-shot, and grounding settings on three datasets using single-prompt models and using ensemble models. We evaluated

---

[3] We repeat here the Equation 9.1, Section 1, for readability.

the models quantitatively based on the question-answering benchmarks and qualitatively with the help of a medical expert. In this Section, we present the gist of the results.

3.1 Experimental setup

Language Model All the main experiments were carried out using InstructGPT, a version of the largest of GPT-3 (175 billion parameters) fine-tuned for prompt-based learning based on human feedback (Ouyang et al. 2022).

Predictions All models were sampled deterministically (greedy decoding) via the OpenAI API.[4] The probability $p_\theta\big(\mathbf{x} \mid f_2([\bar{\mathbf{y}}; \mathbf{z}], \mathbb{K})\big)$ is approximated with 1 when the correct answer appears in the generated answer string, 0 otherwise.

Grounding In the grounded setting, we retrieved $K = 1$ document for each answer option (four documents per question) using a composite BM25 (Robertson and Zaragoza 2009) retriever indexing passages collected from Wikipedia articles.

3.2 Zero-shot answering accuracy

Table 9.3: Zero-shot answering accuracy (%) of GPT-3 on the MedMCQA (validation) and USMLE (test) datasets using CoT-free prompts, the original CoT prompt and an ensemble of prompts, with and without grounding (conditioning on Wikipedia articles).

Model	Grounding	Prompting	MedMCQA	USMLE
GPT-3	✗	Standard	44.0	46.0
GPT-3	✗	CoT*	40.8	47.1
GPT-3	✓	Standard	46.7	47.3
GPT-3	✓	CoT*	42.2	45.9
GPT-3	✗	Ensemble (P=6)	42.4	50.0
GPT-3	✓	Ensemble (P=6)	**48.8**	49.3
GPT-3	✗ + ✓	Ensemble (P=12)	47.6	**53.1**
Uniform			25	25
Fine-tuned SOTA (BERT)	✗	–	40†	–
Fine-tuned SOTA (BERT)	✓	–	43†	44.6‡
Human (passing score)	✓	–	≥50.0	≥60

*"*Let's think step by step*" †Pal, Umapathi, and Sankarasubbu 2022 ‡Yasunaga, Leskovec, and Liang 2022

---

[4] `https://openai.com/api`

In Table 9.3, we summarize the answering performances on the MedMCQA and the USMLE datasets and discuss below the main findings. In Paper C, we provide further empirical results and an extended analysis of the results.

Single-prompt Looking at the single-prompt models, this experiment allowed us to demonstrate that GPT-3 is knowledgeable about expert-level medical questions, as it outperformed all the fine-tuned BERT baselines, even when the baselines are conditioned on external knowledge sources.

Ensemble models Single-prompt CoT models are not necessarily better than the CoT-free models. However, combining multiple prompts overall led to significantly better performances.

Grounding Retrieval-augmented models significantly outperformed the others on the MedMCQA, indicating that grounding GPT-3 with more knowledge can be helpful. Nonetheless, it is not clear whether grounding helps with the USMLE questions; it is likely that the BM25 retriever is too simplistic for this problem.

3.3 Qualitative analysis

Table 9.4: Assessing GPT-3's ability to reason. Frequencies of observed success (A,B,C) and failure (D,E,F) patterns were identified among 50 CoTs. A CoT is labelled if it contains at least one expression of the pattern, therefore a single CoT can be classified both as a reasoning success and a reasoning failure.

	Pattern	Total
A	Correct reasoning step	70%
B	Correct recall of knowledge	72%
C	Correct reading comprehension	90%
D	Incorrect reasoning step	62%
E	Incorrect or insufficient knowledge	58%
F	Incorrect reading comprehension	36%

The question-answering benchmark provides an overall assessment of GPT-3's ability to answer difficult medical questions, In Paper C, we investigated how GPT-3 succeeds and fails. Although language models are known to be difficult to interpret, CoT prompting is a unique opportunity for assessing the strengths and weaknesses of GPT-3. We considered a set of three main competencies: (i) ability to reason, (2) ability to recall knowledge and (3) reading comprehension (find a sample in Table 9.5).

In Table 9.4, we report the frequency of each success and failure for each of the patterns annotated by our medical expert. In Paper C, we display 30 CoT with annotated successes and failures. Overall, GPT-3 appears to be more proficient in reading comprehension than in the reasoning and knowledge categories. Although failures were observed in all categories, the frequency of success patterns is encouraging. In many cases, zero-shot CoT prompting successfully triggered GPT-3 to generate multi-step problem solutions, which mobilize non-trivial reasoning steps, recall of knowledge and correct interpretation of medical data.

4 Conclusion

We demonstrated that, without requiring any domain-specific fine-tuning, GPT-3 can outperform the state-of-the-art fine-tuned language models on challenging medical question-answering datasets. We applied diverse prompting techniques, including zero-shot CoT prompting, which allowed us to interpret the model predictions. The qualitative evaluation indicated that CoTs overall make sense: GPT-3 mostly comprehend the provided data and can often relate it to its own medical knowledge with the aim of solving a problem.

Although Paper C pioneered the use of large language models for answering complex and realistic medical questions, our methodology remains simple. Leveraging more inference-time compute by sampling a larger number of CoTs would allow us to automatically explore many potential solutions to each problem, in turn leading to better medical problem-solving performances.

Our results hint that a lack of domain knowledge could be compensated by coupling language models with external sources of information; language models might lack of domain knowledge but excel at reading comprehension. Therefore, using a more accurate retriever and retrieving more document passages is likely to help. Augmenting language models with retrievers is the topic of the next chapter, in which we present a framework for learning retrievers jointly with a question-answering model.

Table 9.5: USMLE question and three annotated CoTs.

<table>
<tr><td colspan="3">Three days after induction chemotherapy, a 9-year-old boy is admitted to the hospital for fever and neutropenia. He was initially diagnosed with t(9;22)-positive acute T cell lymphoblastic leukemia following workup for fatigue and pancytopenia. He is fully conscious. His blood pressure is 110/65 mm Hg, temperature is 39.5°C (103.1°F), pulse is 98/min, and respiratory rate is 14/min. Physical examination including examination of the central venous catheter site shows no source of infection. Broad-spectrum antibiotics are initiated. Which of the following factors about this patient most strongly indicates a poor prognosis?
A) Age <b>B) Cytogenetics</b> C) Fever and neutropenia D) Initial presentation</td></tr>
<tr><td>✓ <b>Let's use step by step inductive reasoning, given the medical nature of the question</b>.
The first thing that stands out is that the patient is 9 years old. In general, children have a better prognosis than adults when it comes to leukemia.
The next thing to consider is the patient's cytogenetics. The fact that the patient has t(9;22)-positive acute T cell lymphoblastic leukemia indicates that the patient has a specific chromosomal abnormality that is associated with a poor prognosis.
The next thing to consider is the patient's fever and neutropenia. Fever and neutropenia are both signs of infection, and infection is a serious complication in leukemia patients, particularly in those who are immunocompromised. Finally, we need to consider the patient's initial presentation. The fact that the patient was initially diagnosed with leukemia following workup for fatigue and pancytopenia indicates that the patient's leukemia was advanced at the time of diagnosis.
All of these factors indicate a poor prognosis for this patient.
<b>Therefore, among A through D, the answer is B, cytogenetics.</b></td><td>✗ <b>Let's think step by step like a medical expert</b>. The first step is to gather information about the patient. In this case, we are given the following information: -The patient is 9 years old -The patient has t(9;22)-positive acute T cell lymphoblastic leukemia -The patient is currently experiencing fever and neutropenia
The next step is to analyze the information and look for any patterns or clues that could help us answer the question.
In this case, the most important factor that indicates a poor prognosis is the patient's age. Children under the age of 10 have a much higher chance of relapsing after treatment and a lower chance of achieving remission.
<b>Therefore, among A through D, the answer is A.</b></td><td>✗ <b>Let's derive the differential diagnosis step by step</b>: 1) The patient has fever and neutropenia.
2) The patient is 9 years old.

3) The patient has t(9;22)-positive acute T cell lymphoblastic leukemia.
4) The patient's initial presentation was fatigue and pancytopenia.
The most likely cause of the patient's fever and neutropenia is infection. The patient's age, cytogenetics, and initial presentation are all risk factors for infection.
Therefore, the factor that most strongly indicates a poor prognosis is the patient's age.
<b>Therefore, among A through D, the answer is A.</b></td></tr>
</table>

A	Correct reasoning step	D	Incorrect reasoning step
B	Correct recall of knowledge	E	Incorrect or insufficient knowledge
C	Correct reading comprehension	F	Incorrect reading comprehension

CHAPTER 10

Learning to Search Wikipedia by Answering Medical Questions

This chapter cites one of the contributions in this thesis:

Paper D: "Variational Open-Domain Question Answering" (Liévin et al. 2022b)

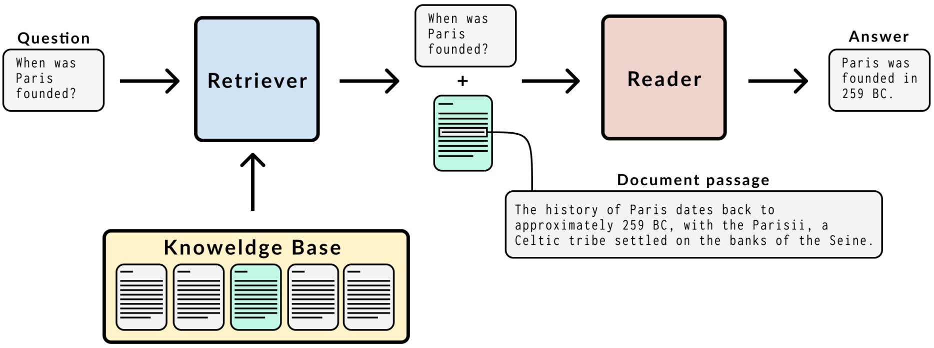

Question
When was Paris founded?
Retriever
When was Paris founded?
+
Reader
Answer
Paris was founded in 259 BC.
Document passage
The history of Paris dates back to approximately 259 BC, with the Parisii, a Celtic tribe settled on the banks of the Seine.
Knoweldge Base

Figure 10.1: Open-domain question answering: answering questions by retrieving knowledge from a large knowledge base.

Large language models have taken over the field of natural language processing. Scaling Transformers to using billions of parameters and pre-training by modelling hundreds of gigabytes of text[1] results in exceptionally good language models. Nonetheless, it is challenging to ensure that all information referenced in the pre-training set is factual, not hateful or discriminative. A failure to curate the training data which leads to large language models repeating, and sometimes amplifying, dangerous content (Bender et al. 2021). Furthermore, language models don't necessarily retain all pre-training knowledge and, at inference-time, models might be subject to

---

[1] The Pile (Gao et al. 2021), a corpus utilized to train large language models, contains 825 GB of text, Wikipedia contains around 33 GB of text.

hallucinating erroneous knowledge (Maynez et al. 2020). There are flaws that need to be controlled before deploying large language models in sensitive applications such as healthcare.

Grounding language models in curated knowledge sources offers a solution to these limitations (Lazaridou et al. 2022; Shuster et al. 2021). Instead of relying solely on the knowledge retained in their weights, language models can be augmented with external knowledge bases indexed by a search component (e.g., search engine, BM25, language model). Knowledge bases, such as Wikipedia, can be more easily curated and updated, ultimately enabling retrieval-augmented language models to be applied to new domains without additional pre-training.

In Paper D, we introduced the Variational Open-Domain (VOD) framework for end-to-end training and evaluation of retrieval-augmented models. We applied the framework to learn retrieval-augmented question answering models with a differentiable search component. Using a fraction of Wikipedia as a knowledge base, our approach resulted in state-of-the-art answering accuracy on two medical question-answering datasets. The search component learned in the process appeared to be competitive with FindZebra; a search engine specialized for rare diseases.[2] In this chapter, we begin with providing background on open-domain question answering before introducing the Variational Open-Domain (VOD) framework for end-to-end learning of retrieval-augmented models using variational inference. We conclude this chapter with an overview of our empirical findings.

1 Background

In this section, we introduce the main building blocks of open-domain question answering and retrieval-augmented language modelling (task definition, parameterization of the models using pre-trained language models and optimization objectives). Readers with expertise in open-domain question answering can safely skip this section and resume reading at Section 2.

1.1 Open-Domain Question Answering

Open-domain question answering (ODQA) is the task of answering questions using an unstructured knowledge base such as Wikipedia. The task originates from the annual competition held at the Text REtreival Conference (TREC)[3], which catalyzed advances in ODQA, and ultimately led to the creation of complex ODQA systems like YodaQA (Baudiš 2015) and DeepQA(Ferrucci et al. 2010); the underlying system that led to the much-publicised victory of IBM Watson in the Jeopardy! challenge[4].[5]

---

[2] `http://findzebra.com`

[3] `http://trec.nist.gov/data/qamain.html`

[4] `http://en.wikipedia.org/wiki/IBM_Watson`

[5] Credits to Rajpurkar et al. 2016 and Chen et al. 2017a for summarizing the history of ODQA.

Classical ODQA systems like YodaQA or DeepQA answer questions by combining information from multiple structured (Freebase, DBPedia) and unstructured sources (Wikipedia) using multiple layers of expert systems. Chen et al. 2017a leveraged the recent progress in natural language understanding to design a much simpler system. This system, dubbed DrQA, can answer questions relying on an unstructured knowledge base (Wikipedia) and two components: a TF-IDF retriever and a machine reading comprehension model to answer questions based on a context document.

1.2 Latent retrieval

Instead of treating the document retriever as a static component, Lee, Chang, and Toutanova 2019 views ODQA as an instance of a conditional latent variable model. Given a question $\mathbf{q}$ with answer $\mathbf{a}$ and a corpus of documents $\mathcal{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_N]$, the marginal task likelihood is expressed as

$$p_\theta(\mathbf{a} \mid \mathbf{q}) := \sum_{\mathbf{d} \in \mathcal{D}} \underbrace{p_\theta(\mathbf{a} \mid \mathbf{q}, \mathbf{d})}_{\text{reader}} \underbrace{p_\theta(\mathbf{d} \mid \mathbf{q})}_{\text{retriever}} , \tag{10.1}$$

where $p_\theta(\mathbf{d} \mid \mathbf{q})$ is potentially a learned distribution implemented using neural networks (retriever), $p_\theta(\mathbf{a} \mid \mathbf{q}, \mathbf{d})$ is a machine reading comprehension model (reader) and θ is the parameter describing the whole system with joint distribution $p_\theta(\mathbf{d}, \mathbf{a} \mid \mathbf{q})$.

1.3 Differentiable Search

In the literature, the retriever is defined as a truncated distribution

$$p_\theta(\mathbf{d} \mid \mathbf{q}) := \frac{\mathbb{1}[\mathbf{d} \in \mathcal{T}_\phi(\mathbf{q}; K)] \exp f_\theta(\mathbf{q}, \mathbf{d})}{\sum_{\mathbf{d}' \in \mathcal{T}_\phi(\mathbf{q}; K)} \exp f_\theta(\mathbf{q}, \mathbf{d}')}, \tag{10.2}$$

where $f_\theta(\mathbf{q}, \mathbf{d})$ scores the document $\mathbf{d}$ given the query $\mathbf{q}$ and $\mathcal{T}_\phi(\mathbf{q}; K)$ is a set of the top K documents given a scoring function f_ϕ with argument $\mathbf{q}$. K is the number of documents that fit in memory when evaluating the training objective. In general, most of the documents in the corpus have almost zero probability under $p_\theta(\mathbf{d}|\mathbf{q}) \propto \exp f_\theta(\mathbf{q}, \mathbf{d})$, thus f_ϕ is generally defined as a checkpoint of f_θ.[6]

The scoring function f_θ can be implemented using pre-trained language models. However, computational efficiency is a key requirement, as collections of documents might reference million or even trillion of documents (Borgeaud et al. 2021). We consider a BERT pre-trained language model (Devlin et al. 2019) and denote $\textsc{bert}(\mathbf{x})$ the vector representation of $\mathbf{x}$ returned by BERT (output at the `CLS` token).[7] Below, we present two alternatives to implementing $f_\theta(\mathbf{q}, \mathbf{d})$ using language models.

---

[6] In the literature, the truncated retriever distribution is defined implicitly by assuming a top-K approximation. The truncated re-interpretation 10.2 is a contribution of Paper D.

[7] Read more about BERT in Section 2

Dual-encoder In most of the ODQA methods, the scoring function adopts a dual-encoder architecture. This consists of encoding $\mathbf{q}$ and $\mathbf{d}$ into vector representations $\textsc{bert}(\mathbf{q})$ and $\textsc{bert}(\mathbf{d})$ separately and defining the scoring function using a vector similarity metric. In the case of the inner product, the scoring function is defined as:

$$f_\theta(\mathbf{q}, \mathbf{d}) = \textsc{bert}(\mathbf{q})^T \textsc{bert}(\mathbf{d}) \ . \tag{10.3}$$

Each vector representation is evaluated independently of the other variables. Thus, for a dataset of $N_\mathbf{q}$ queries and $N_\mathbf{d} = N$ documents, computing all vector representations is of order $\mathcal{O}(N_\mathbf{q} L_\mathbf{q} + N_\mathbf{d} L_\mathbf{d})$ where $L_\mathbf{q}$ and $L_\mathbf{d}$ are the maximum number of query and document tokens. Retrieving the set $\mathbb{T}_\phi(\mathbf{q}; K)$ of the top-K documents can be achieved with low-latency thanks to efficient vector similarity search libraries like `faiss` (Johnson, Douze, and Jegou 2021). In practice, the document vectors are fixed (Borgeaud et al. 2021; Lewis et al. 2020), evaluated asynchronously (Guu et al. 2020) or periodically (Liévin et al. 2022b; Paranjape et al. 2021).

Cross-attention encoder The inner-product scoring model defined in Equation 10.3 might be restrictive. The full machine reading comprehension capabilities of the language model can be exploited by processing both the question and the document as a single input $[\mathbf{d}; \mathbf{q}]$, given a projection matrix W with output dimension 1:

$$f_\theta(\mathbf{q}, \mathbf{d}) = \textsc{bert}\left([\mathbf{d}; \mathbf{q}]\right) W \ . \tag{10.4}$$

Nonetheless, the computational computational complexity of the cross-attention model is $\mathcal{O}(N_\mathbf{q} N_\mathbf{d} (L_\mathbf{q} + L_\mathbf{d}))$ and is therefore prohibitively expensive to evaluate. Cross-attention encoders are generally coupled with a cheaper retrieval process and used as a re-ranking step. This was explored in Lazaridou et al. 2022 by re-ranking Google Search results thanks to a large language model.

1.4 Learning to retrieve documents

Weak supervision In the case where document labels are available, or can be inferred accurately enough, a differentiable retriever can be optimized via supervised learning, as originally explored in Karpukhin et al. 2020 (Dense passage retrieval).

Likelihood-based learning The marginal likelihood defined in Equation 10.1 is differentiable and therefore the whole system can be optimized jointly. However, the Monte Carlo estimate of the gradient of Equation 10.1 often suffers from high variance at initialization, limiting its feasibility.

Lower variance with pre-training The problem of high variance can be contained by pre-training the retriever using an auxiliary task such as the *Inverse Cloze Task* (ICT, Lee, Chang, and Toutanova 2019). The ICT task consists in learning to predict whether two sentences belong to the same context, which has been successfully applied as an unsupervised pre-training scheme in subsequent work (Guu et al. 2020; Izacard et al. 2021; Sachan et al. 2021).

Variational inference Alternatively ODQA models can be trained using a variational objective. This corresponds into using an approximate posterior $r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$ which can be conditioned on the answer $\mathbf{a}$. In Paranjape et al. 2021, the parameters of the system are maximized using both marginal task likelihood 10.1 and the evidence lower bound:

$$\mathcal{L}(\mathbf{a}, \mathbf{q}) := \mathbb{E}_{q_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[\log \frac{p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q})}{r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})}\right] . \tag{10.5}$$

2 The Variational Open-Domain Framework

In this section, we present the *Variational Open-Domain* (VOD) framework, which consists of a variational re-interpretation of ODQA backed with computationally efficient importance-sampling estimates. We begin with presenting the Rényi variational bound for ODQA, discuss the choice of retriever and approximate posterior and conclude presenting with presenting tractable self-normalized importance sampling estimates for the variational bound.

2.1 Rényi variational bound

VOD estimates the log marginal task likelihood using the *Rényi variational bound* (RVB, see section 4). The RVB is defined based on an answer-aware approximate posterior $r_\phi(\mathbf{a} \mid \mathbf{a}, \mathbf{q})$. Using a parameter $\alpha \neq 1$ and defining $w_{\theta,\phi}(\mathbf{q}, \mathbf{a}, \mathbf{d}) := p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q}) / r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$, the RVB for ODQA is:

$$\mathcal{L}_\alpha(\mathbf{a}, \mathbf{q}) = \frac{1}{1-\alpha} \log \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[w_{\theta,\phi}^{1-\alpha}(\mathbf{q}, \mathbf{a}, \mathbf{d})\right] . \tag{10.6}$$

The RVB is a lower bound of the marginal task log-likelihood for $\alpha \geq 0$ its definition is extended in $\alpha = 1$ by continuity using $\mathcal{L}_{\alpha=1}(\mathbf{a}, \mathbf{q}) := \lim_{\alpha \to 1} \mathcal{L}_\alpha(\mathbf{a}, \mathbf{q})$, which corresponds to the ELBO $\mathcal{L}(\mathbf{a}, \mathbf{d})$ (Equation 10.5). The RVB is continuous in α and allows connecting the marginal likelihood and the ELBO.

2.2 Retrievers with extended document support

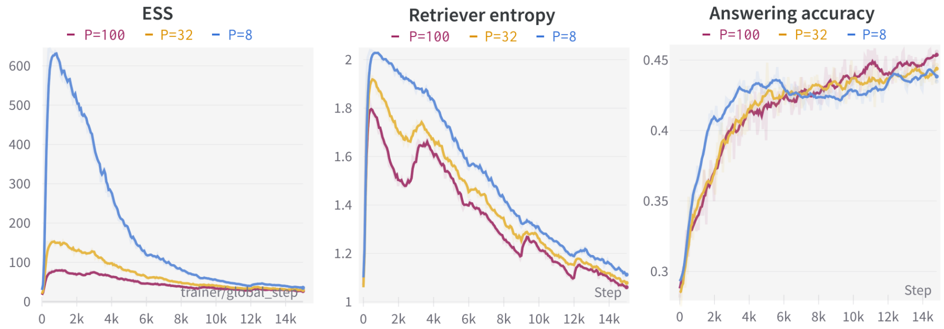


Figure 10.2: Effect of the choice of P (size of the retriever support) on the training of multiple-choice ODQA models. The approximate posterior depends on a checkpoint of the retriever, updated every 3k steps. Using Larger values of P leads to lower ESS, slower learning but ultimately better validation answering accuracy. **NB** The retriever is defined for the multiple-choice setting with $p_\theta(\mathbf{D}|\mathbf{Q}) = \prod_{i=1}^{M} p_\theta(\mathbf{d}_i|\mathbf{q}_i)$ and $M = 4$. Sampling $K = 8$ documents for each of the M answer options yields $K^M = 4096$ unique document tuples $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3, \mathbf{d}_4]$ and as many importance weights.

The approximate posterior $r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$ is a parameterized by a function $f_\phi(\mathbf{q}, \mathbf{a}, \mathbf{d})$. The framework doesn't impose a choice of f_ϕ; in Paper D, f_ϕ is constructed as a mixture of a checkpoint of f_θ and a BM25 baseline.

Our work stands out of from the literature because it allows extending the size of the support of the retrieval distribution to $P > K$, even if P documents cannot be evaluated with a single call due to hardware limitations. Denoting $\mathbb{T}_\phi := \mathcal{T}_\phi(\mathbf{q}, \mathbf{a}; P)$ the corresponding set of the top P documents ranked by f_ϕ, in our work, the retriever and the approximate posterior are defined as

$$p_\theta(\mathbf{d} \mid \mathbf{q}) := \frac{\mathbb{1}[\mathbf{d} \in \mathbb{T}_\phi] \exp f_\theta(\mathbf{q}, \mathbf{d})}{\sum_{\mathbf{d}' \in \mathbb{T}_\phi} \exp f_\theta(\mathbf{q}, \mathbf{d}')}, \qquad r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q}) := \frac{\mathbb{1}[\mathbf{d} \in \mathbb{T}_\phi] \exp f_\phi(\mathbf{q}, \mathbf{a}, \mathbf{d})}{\sum_{\mathbf{d}' \in \mathbb{T}_\phi} \exp f_\phi(\mathbf{q}, \mathbf{a}, \mathbf{d}')}. \tag{10.7}$$

Although choosing small values of P comes with a minor technical advantage (reducing disk or memory usage), the choice of P has a more significant effect on the optimization behaviour. The size of the set $\mathbb{T}_\phi$ defines a trade-off between exploration and exploitation: a higher value of P allows sampling a greater diversity of documents (*exploration*: high variance), but a smaller value makes it more likely that all documents in the set $\mathbb{T}_\phi$ will be visited during training (*exploitation*: low variance). In Figure 10.2, we report the training curves for multiple-choice ODQA models for $P \in \{8, 32, 100\}$ and $K = 8$.

2.3 Monte Carlo estimation

Based on a retriever restricted to the top-K documents, the marginal likelihood 10.1 can be evaluated directly. Using the top-P retriever parameterization 10.7, estimating the variational bound 10.6 poses three challenges, which we list below along with practical solutions.

1. **RVB estimation** The RVB defined in 10.6 is intractable, it is estimated using the K-sample importance weighted Rényi bound denoted $\mathcal{L}_\alpha^K(\mathbf{x})$ (introduced in Section 3, Equation 3.17):
$$\mathcal{L}_\alpha^K(\mathbf{a},\mathbf{q}) := \frac{1}{1-\alpha}\mathbb{E}_{q_\phi(\mathbf{d}_{1:K}|\mathbf{a},\mathbf{q})}\left[\log\frac{1}{K}\sum_{i=1}^{K} w_{\theta,\phi}^{1-\alpha}(\mathbf{a},\mathbf{q},\mathbf{d}_i)\right] \leq \mathcal{L}_\alpha(\mathbf{a},\mathbf{q})\ . \quad (10.8)$$
2. **Retriever normalizing constant estimation** The normalizing constant $\sum_{\mathbf{d}\in\mathbb{T}_\phi}\exp f_\phi(\mathbf{q},\mathbf{d}')$ of the retriever distribution 10.7 is intractable as it requires $P > K$ evaluations of f_θ. Using the ratio of scores $\zeta(\mathbf{d}) := \frac{\exp f_\theta(\mathbf{q},\mathbf{d})}{\exp f_\phi(\mathbf{q},\mathbf{a},\mathbf{d})}$, we estimate the normalizing constant using the equality:
$$\frac{\sum_{\mathbf{d}\in\mathbb{T}_\phi}\exp f_\theta(\mathbf{q},\mathbf{d})}{\sum_{\mathbf{d}'\in\mathbb{T}_\phi}\exp f_\phi(\mathbf{q},\mathbf{a},\mathbf{d}')} = \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[\zeta(\mathbf{d})\right]\ . \quad (10.9)$$
3. **Optimal resource allocation** Each evaluation of $f_\theta(\mathbf{q},\mathbf{d})$ is expensive and the most of the mass of $r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})$ might be concentrated in a few documents. To avoid evaluating the same document multiple times, we applied *self-normalized priority sampling* (see Section 1.3). Given $\mathbb{S} = \{\mathbf{d}_1,\ldots,\mathbf{d}_K\} \subset \mathbb{T}_\phi$ a set of documents sampled without replacement from $r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})$, priority sampling comes with self-normalized weights $\tilde{s}(\mathbf{d}_1),\ldots,\tilde{s}(\mathbf{d}_K)$ defined such that for a function $h(\mathbf{d})$,
$$\sum_{\mathbf{d}\in\mathbb{S}}\tilde{s}(\mathbf{d})h(\mathbf{d}) \approx \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[h(\mathbf{d})\right]\ . \quad (10.10)$$

Combining the three above parts (Equations 10.8, 10.9 and 10.10), we obtain a tractable importance sampling estimate of the RVB, which is the training and evaluation bound given by the VOD framework:

$$\mathcal{L}_\alpha^K(\mathbf{a},\mathbf{q}) \approx \frac{1}{1-\alpha}\log\sum_{\mathbf{d}\in\mathbb{S}}\tilde{s}(\mathbf{d})\left(\frac{p_\theta(\mathbf{a}\mid\mathbf{d},\mathbf{q})\zeta(\mathbf{d})}{\sum_{\mathbf{d}'\in\mathbb{S}}\tilde{s}(\mathbf{d}')\zeta(\mathbf{d}')}\right)^{1-\alpha}\ . \quad (10.11)$$

In Paper D, we provide the derivation and an estimate for the gradient of the generative model. Equation 10.11 is an instance of self-normalized importance sampling (see Section 1) and is therefore consistent (i.e., converge to the true expected value in the limit $K = P$ with probability one). In our work, the approximate posterior is considered static and defined based on a checkpoint of the retriever and BM25. Thus, we didn't attempt optimizing the variational parameter but the OVIS gradient estimators introduced in Paper B are applicable to the VOD framework.

3 Main Results

In Paper D, we applied the framework to the task of multiple-choice ODQA and evaluated the learned retriever based on an information retrieval benchmark.[8] Applying the VOD framework to multiple-choice QA requires some adaptation. The gist of it consists of using a multiple-choice reader paired with a per-option retriever:

$$p_\theta(\mathbf{a}_\star \mid \mathbf{D}, \mathbf{Q}) := \frac{\exp g_\theta(\mathbf{d}_\star, \mathbf{q}_\star)}{\sum_{j=1}^{M} \exp g_\theta(\mathbf{d}_j, \mathbf{q}_j)}, \qquad p_\theta(\mathbf{D} \mid \mathbf{Q}) := \prod_{j=1}^{M} p_\theta(\mathbf{d}_j \mid \mathbf{q}_j), \tag{10.12}$$

where $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_M]$ is a tuple of M answer options, $\star$ is the index of the correct answer, g_θ is a function parameterized by BERT and $\mathbf{D}$ is tuple of M documents and $\mathbf{Q} = [[\mathbf{q}; \mathbf{a}_1], \ldots, [\mathbf{q}; \mathbf{a}_M]]$. In this section, we present an overview of the main empirical findings: good ODQA performances and applicability to information retrieval.

3.1 Multiple-choice ODQA

Table 10.1: ODQA accuracy of an ODQA models VOD framework. All models index Wikipedia articles. VOD models are evaluated using Equation 10.11 with C=10 Monte Carlo samples, each containing M·K=32 documents. GPT-3 is evaluated with M·K=4.

			MedMCQA		**USMLE**	
Method	**Reader**	**Retriever**	**Valid.**	**Test**	**Valid.**	**Test**
VOD	BioLinkBERT†	BM25	51.6	55.3	41.0	40.4
VOD	BioLinkBERT†	BioLinkBERT†	**58.3**	**62.9**	**53.6**‡	**55.6**‡
Uniform baseline	–	–	25.0	25.0	25.0	25.0
Zero-shot prompting	GPT-3	–	44.0*	–	–	46.0
Zero-shot prompting	GPT-3	BM25	46.7*	–	–	47.3
Zero-shot CoT prompting	GPT-3	BM25	48.8*	–	–	53.1
Human (passing score)	–	–	≥ 50	≥ 50	≥ 60	≥ 60

*Estimated using 1k samples †Yasunaga, Leskovec, and Liang 2022 ‡Pre-training on the MedMCQA dataset

In Table 10.1, we report the answer accuracy of multiple models on the validation and test sets on the medical QA datasets introduced in Chapter 9, Section 1. Compared to an ODQA system with a static retriever component (BM25), learning a retriever jointly with the reader component using the VOD framework leads to significantly better answering performances on both datasets, outperforming the strong GPT-3 reasoner discussed in Chapter 9. In the next section, we study the learned retriever from an information retrieval standpoint.

---

[8] Find a detailed description of the experimental setup in Paper D.

3.2 Information retrieval

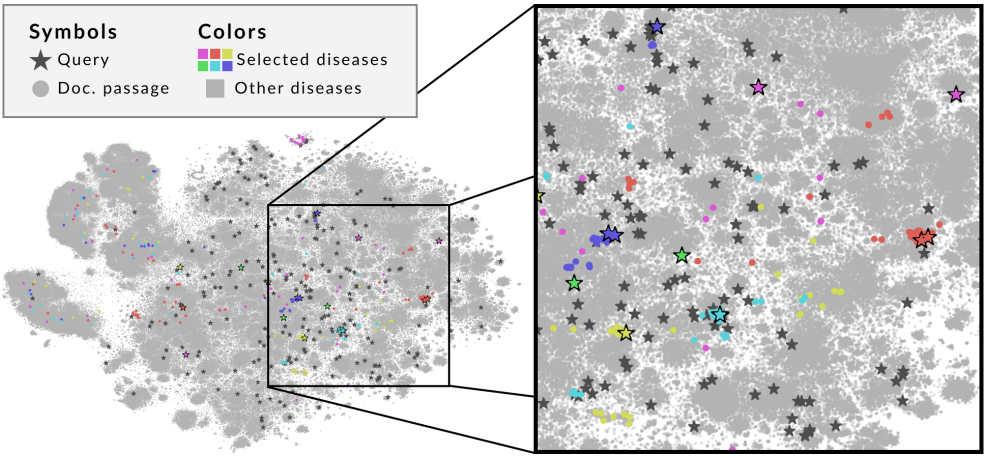


Figure 10.3: Visualizing the latent retrieval space. T-SNE projection of the embedding space where are encoded the 712k document passages of the FindZebra corpus and the 248 FindZebra queries. The documents and questions are annotated based on their disease identifier. The documents and queries annotated with the top 6 most frequent diseases (found in the queries) are highlighted with colours. The others are represented in gray. Some queries are successfully matched with a neighbourhood of relevant passages, although passages taken from a single document might be scattered across the embedding space.

FindZebra is a search engine aiming at assisting the medical professionals in the diagnosis of rare diseases (Dragusin et al. 2013).[9] The search engine indexes a corpus of 30.7k high-quality curated articles using Apache Solr (specialized BM25). In Paper C, we compared our approach with the specialized tool based on a set of 248 search queries annotated with the correct diagnosis. The task consists of retrieving a document matching the diagnosis associated with the query. Searching documents using a BERT-based retriever translates into a nearest neighbour search problem in the embedding space, which we visualize in Figure 10.3.

Task adaptation using distillation A multiple-choice retriever, as defined in Equation 10.7, learns a retrieval function that depends both on the question $\mathbf{q}$ and the answer option $\mathbf{a}_i$. We experimented with adapting the retriever to retrieve documents based only on the question $\mathbf{q}$ via knowledge distillation (Hinton, Vinyals, and Dean 2015). This corresponds to optimizing a retriever $p_\theta(\mathbf{d} \mid \mathbf{q})$ by minimizing the divergence $D_{\mathrm{KL}}(r_\phi(\mathbf{d}|[\mathbf{q};\mathbf{a}]) \,||\, p_\theta(\mathbf{d}|\mathbf{q}))$ where $r_\phi(\mathbf{d}|[\mathbf{q};\mathbf{a}])$ is a trained multiple-choice VOD retriever.

---

[9] `https://www.findzebra.com/`

Table 10.2: Retrieval performances. Comparing deep retrievers with FindZebra and a simple BM25 baseline. the performances are reported with and without task adaptation (distillation).

Method	**Distil.**	**MRR**	**Hit@20**
VOD	✗	27.8	56.9
VOD	✓	31.7	58.1
VOD+BM25	✓	**38.9**	**64.1**
BM25	–	26.4	48.4
FindZebra	–	30.1	59.3

Results In Table 10.2, we compared the learned deep retrievers with API FindZebra based on two popular metrics in the field of information retrieval and the mean reciprocal rank (MRR) and the proportion of questions which correct article appears in the top-20 results (Hit@20). We found that with further task adaptation, VOD allowed learning retrievers that are competitive with a specialized search tool. When coupled with a simple lexical search baseline (BM25) our approach outperforms the FindZebra API.

4 Conclusion

VOD is a new framework for evaluating and end-to-end training of retrieval augmented models. The Rényi variational bound introduces a trade-off between the tightness of the lower bound and overcoming optimization challenges; as studied in Chapter 8. Compared to the literature, VOD is the first method that applies variational inference methodically, allowing defining document retrievers on arbitrary subsets of documents ($P \geq K$). We hope VOD will spark more interest in applying variational inference in natural language processing problems.

The multiple-choice VOD end-to-end systems scored new state-of-the-art results on two medical question-answering datasets, outperforming the method presented in Chapter C. Given the impressive reading comprehension capabilities of GPT-3, this indicates that VOD allows learning strong document retrievers. We confirmed this intuition by challenging the specialized rare disease search engine FindZebra.

Applying VOD to optimize differentiable ODQA systems resulted in a significant performance jump compared to the literature. This result corroborates the good performances of retrieval-augmented models found in the literature. We hope that VOD will contribute to large language models pushing the current boundaries of machine intelligence.

CHAPTER 11

Conclusion

1 Summary

This thesis investigates the use of deep latent variable models for natural language processing. We organized the dissertation on three following research questions:

- **Q1:** How to design deep latent variable models for text data?
- **Q2:** How to optimize latent variable models using variational inference?
- **Q3:** How to apply latent variable models to natural language processing tasks?

We first review background material in Monte Carlo methods, variational inference and deep generative models. In the second part, we discussed three types of latent variable models for text, two of which relied on language as a latent variable. In the third part, we delved into the topic of variational inference and optimization. Finally, we experimented with the tasks of medical question-answering and information retrieval, developing domain-specific methods in the process.

Deep latent variable models for language modelling (**Q1**) In chapter 5, we introduced BIVA: a deep Variational Autoencoder designed with a flexible bidirectional *bottom-up* and *top-down* inference network (Paper A). We presented the bidirectional architecture and reviewed one of the image modelling experiments (CIFAR-10). showing that BIVA's deep hierarchy of latent variables is competitive with the autoregressive models. We Applied BIVA to text modelling and reported that autoregressive language components might be a necessary component for text modelling.

In Chapter 6, we presented two models with structured latent spaces. The first model (Paper C) defines a latent *reasoning* process. The second model (Paper D) is a language model augmented with a retrieval process that indexes a large collection of documents (open-domain question answering).

Importance weighted bounds and optimization (**Q2**) In chapter 7, we sought the optimal control variate for the score function estimator applied to the importance weighted bound (Paper B). We presented a new family of estimators named OVIS. It extends the score function estimator (and VIMCO) with more optimal baselines. The type of baseline must be selected based on the task (compute budget, effective sample size). We concluded the chapter with a discussion about the limitation of the importance weighted bound in the low effective sample size regime.

In Chapter 8, based on empirical results from Papers B and A, we showed that Rényi divergence variational inference can be applied to circumvent the optimization challenges associated with the importance weighted bound. In practice, this consists of interpolating between the evidence lower bound and the importance weighted bound.

Medical question-answering and information retrieval (**Q3**) In Chapter 9, we applied large language models (GPT-3) to answer challenging medical questions. We engineered prompts to trigger the language model to generate chain of thoughts, to utilize factual documents retrieved from Wikipedia (grounding) and to utilize reference question-answer pairs (few-shot learning). We recorded good answering performances, notably when combining multiple chain-of-thoughts.

In chapter 10, we presented VOD (*Variational Open-Domain*), a framework for end-to-end training and evaluation of retrieval-augmented language models. The framework consists of using the variational Rényi bound, defining truncated retriever parameterization and estimating the bound and its gradient using self-normalized priority sampling. We applied the framework to learn end-to-end open-domain question-answering models and advanced the state-of-the-art on two medical question-answering datasets. We employed the learned retriever for the information retrieval and outperformed the search engine FindZebra on a disease diagnosis benchmark.

2 Closing remarks

In this thesis, we have studied various alternatives to designing latent variable models for natural language processing. Structured latent spaces that encode external data or model generative sub-routines are a promising research direction. Latent variable models can be designed as complex simulators which interact with external knowledge sources using large language models. How to best design and apply such models remains an open research question, but discrete stochastic optimization is likely to play a key role in that domain.

We have studied the topic of variational inference and observed that each variational bound is best utilized depending on the experimental parameters such as the compute budget and the fit of the proposal distribution, which can, for instance, be measured by the effective sample size. Our results corroborate that *"Tighter Variational Bounds are Not Necessarily Better"* (Rainforth et al. 2018), and therefore it is often recommended to use a combination of multiple variational bounds. We found in Rényi divergence variational inference (Li and Turner 2016) an elegant framework to do so. Our findings are aligned with the original philosophy of variational inference, that augmenting the model with additional parameters simplifies the learning problem (Jordan et al. 1999). Maximizing the evidence lower bound guides optimization with constraints imposed by the approximate posterior, leading to a simpler and lower-variance initial objective. Combined with OVIS, this thesis offers a complete method to train deep discrete latent variable models using variational inference.

Large language models are transforming the field of natural language processing and, more generally, machine intelligence. Language models have acquired such strong natural language understanding capabilities that language itself can be used as an interface for machines to learn. In this thesis, this was demonstrated by the zero-shot prompt-based learning performances of GPT-3 in the medical domain. Nonetheless, prompt-based learning is in full bloom. The *pre-train, fine-tune and predict* paradigm is slowly being outpaced by the *pre-train, prompt and predict* (Liu et al. 2021) leading to question the place of variational inference in the future of natural language processing. However, as the costs of fine-tuning large language models will decrease, we might expect the fine-tuning paradigm to burgeon again. In that case, we advocate using VOD to train retrieval-augmented large language models end-to-end, which might pave the way towards better information systems and result in better search engines in the process.

APPENDIX A

BIVA: A Very Deep Hierarchy of Latent Variables for Generative Modeling

BIVA: A Very Deep Hierarchy of Latent Variables for Generative Modeling

Lars Maaløe
Corti
Copenhagen
Denmark
`lm@corti.ai`

Marco Fraccaro
Unumed
Copenhagen
Denmark
`mf@unumed.com`

Valentin Liévin & Ole Winther
Technical University of Denmark
Copenhagen
Denmark
`{valv,olwi}@dtu.dk`

Abstract

With the introduction of the variational autoencoder (VAE), probabilistic latent variable models have received renewed attention as powerful generative models. However, their performance in terms of test likelihood and quality of generated samples has been surpassed by autoregressive models without stochastic units. Furthermore, flow-based models have recently been shown to be an attractive alternative that scales well to high-dimensional data. In this paper we close the performance gap by constructing VAE models that can effectively utilize a deep hierarchy of stochastic variables and model complex covariance structures. We introduce the Bidirectional-Inference Variational Autoencoder (BIVA), characterized by a skip-connected generative model and an inference network formed by a bidirectional stochastic inference path. We show that BIVA reaches state-of-the-art test likelihoods, generates sharp and coherent natural images, and uses the hierarchy of latent variables to capture different aspects of the data distribution. We observe that BIVA, in contrast to recent results, can be used for anomaly detection. We attribute this to the hierarchy of latent variables which is able to extract high-level semantic features. Finally, we extend BIVA to semi-supervised classification tasks and show that it performs comparably to state-of-the-art results by generative adversarial networks.

1 Introduction

One of the key aspirations in recent machine learning research is to build models that *understand the world* [24, 40, 11, 57]. Generative models are providing the means to learn from a plethora of unlabeled data in order to model a complex data distribution, e.g. natural images, text, and audio. These models are evaluated by their ability to *generate* data that is similar to the input data distribution from which they were trained on. The range of applications that come with generative models are vast, where audio synthesis [55] and semi-supervised classification [38, 31, 44] are examples hereof. Generative models can be broadly divided into explicit and implicit density models. The generative adversarial network (GAN) [11] is an example of an implicit model, since it is not possible to procure a likelihood estimation from this model framework. The focus of this research is instead within explicit density models, for which a tractable or approximate likelihood estimation can be performed.

The three main classes of powerful explicit density models are autoregressive models [26, 57], flow-based models [8, 9, 21, 16], and probabilistic latent variable models [24, 40, 33]. In recent years autoregressive models, such as the PixelRNN and the PixelCNN [57, 45], have achieved superior likelihood performance and flow-based models have proven efficacy on large-scale natural image generation tasks [21]. However, in the autoregressive models, the runtime performance of generation is scaling poorly with the complexity of the input distribution. The flow-based models do not possess

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

this restriction and do indeed generate visually compelling natural images when sampling close to the mode of the distribution. However, generation from the actual learned distribution is still not outperforming autoregressive models [21, 16].

Probabilistic latent variable models such as the variational auto-encoder (VAE) [24, 40] possess intriguing properties that are different from the other classes of explicit density models. They are characterized by a posterior distribution over the latent variables of the model, derived from Bayes' theorem, which is typically intractable and needs to be approximated. This distribution most commonly lies on a low-dimensional manifold that can provide insights into the internal representation of the data [1]. However, the latent variable models have largely been disregarded as powerful generative models due to *blurry* generations and poor likelihood performances on natural image tasks. [27, 10], amongst others, attribute this tendency to the usage of a similarity metric in pixel space. Contrarily, we attribute it to the lack of overall model expressiveness for accurately modeling complex input distributions, as discussed in [59, 41].

There has been much research into explicitly defining and learning more expressive latent variable models. Here, the complementary research into learning a covariance structure through a framework of normalizing flows [39, 52, 23] and the stacking of a hierarchy of latent variables [4, 37, 31, 50] have shown promising results. However, despite significant improvements, the reported performance of these models has still been inferior to their autoregressive counterparts. This has spawned a new class of explicit density models that adds an autoregressive component to the generative process of a latent variable model [14, 5]. In this combination of model paradigms, the latent variables can be viewed as merely a *lossy* representation of the input data and the model still suffers from the same issues as autoregressive models.

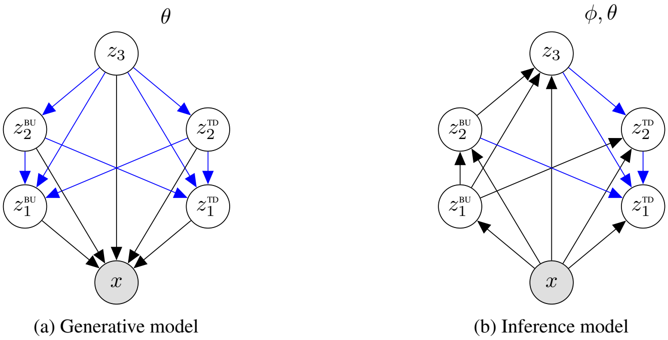
Contributions. In this research we argue that latent variable models that are defined in a sufficiently expressive way can compete with autoregressive and flow-based models in terms of test log-likelihood and quality of the generated samples. We introduce the Bidirectional-Inference Variational Autoencoder (BIVA), a model formed by a deep hierarchy of stochastic variables that uses skip-connections to enhance the flow of information and avoid inactive units. To define a flexible posterior approximation, we construct a bidirectional inference network using stochastic variables in a bottom-up and a top-down inference path. The inference model is reminiscent to the stochastic top-down path introduced in the Ladder VAE [50] and IAF VAE [50] with the addition that the bottom-up pass is now also stochastic and there are no autoregressive components. We perform an in-depth analysis of BIVA and show **(i)** an ablation study that analyses the contributions of the individual novel components, **(ii)** that the model is able to improve on state-of-the-art results on benchmark image datasets, **(iii)** that a small extension of the model can be used for semi-supervised classification and performs comparably to current state-of-the-art models, and **(iv)** that the model, contrarily to other state-of-the-art explicit density models [34], can be utilized for anomaly detection on complex data distributions.

2 Variational Autoencoders

The VAE is a generative model parameterized by a neural network θ and is defined by an observed variable x that depends on a hierarchy of stochastic latent variables $\mathbf{z} = z_1, ..., z_L$ so that: $p_\theta(x, \mathbf{z}) = p_\theta(x|z_1)p_\theta(z_L)\prod_{i=1}^{L-1} p_\theta(z_i|z_{i+1})$. The posterior distribution over the latent variables of a VAE is commonly analytically intractable, and is approximated with a variational distribution which is factorized with a bottom-up structure, $q_\phi(\mathbf{z}|x) = q_\phi(z_1|x)\prod_{i=1}^{L-1} q_\phi(z_{i+1}|z_i)$, so that each latent variable is conditioned on the variable below in the hierarchy. The parameters θ and ϕ can be optimized by maximizing the *evidence lower bound* (ELBO)

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(\mathbf{z}|x)}\left[\log \frac{p_\theta(x, \mathbf{z})}{q_\phi(\mathbf{z}|x)}\right] \equiv \mathcal{L}(\theta, \phi) \ . \tag{1}$$

A detailed introduction on VAEs can be found in appendix A in the supplementary material. While a deep hierarchy of latent stochastic variables will result in a more expressive model, in practice the top stochastic latent variables of standard VAEs have a tendency to *collapse* into the prior. The Ladder VAE (LVAE) [50] is amongst the first attempts towards VAEs that can effectively leverage multiple layers of stochastic variables. This is achieved by parameterizing the variational approximation with a *bottom-up* deterministic path followed by a *top-down* inference path that shares parameters with



(a) Generative model (b) Inference model

Figure 1: A $L = 3$ layered BIVA with (a) the generative model and (b) inference model. Blue arrows indicate that the deterministic parameters are shared between the inference and generative models. See Appendix B for a detailed explanation and a graphical model that includes the deterministic variables.

the top-down structure of the generative model: $q_{\phi,\theta}(\mathbf{z}|x) = q_\phi(z_L|x) \prod_{i=1}^{L-1} q_{\phi,\theta}(z_i|z_{i+1}, x)$. See Appendix A for a graphical representation of the LVAE inference network. Thanks to the bottom-up path, all the latent variables in the hierarchy have a deterministic dependency on the observed variable x, which allows data-dependent information to skip all the stochastic variables lower in the hierarchy (Figure 5d in Appendix A). The stochastic latent variables that are higher in the hierarchy will therefore receive less noisy inputs, and will be empirically less likely to collapse. Despite the improvements obtained thanks to the more flexible inference network, in practice LVAEs with a very deep hierarchy of stochastic latent variables will still experience variable collapse. In the next section we will introduce the Bidirectional-Inference Variational Autoencoder, that manages to avoid these issues by extending the LVAE in 2 ways: (i) adding a deterministic top-down path in the generative model and (ii) defining a factorization of the latent variables z_i at each level of the hierarchy that allows to construct a bottom-up *stochastic* inference path.

3 Bidirectional-Inference Variational Autoencoder

In this section, we will first describe the architecture of the Bidirectional-Inference Variational Autoencoder (Figure 1), and then provide the motivation behind the main ideas of the model as well as some intuitions on the role of each of its novel components. Finally, we will show how this model can be used for a novel approach to detecting anomalous data.

3.1 Model architecture

Generative model. In BIVA, at each layer $1, ..., L-1$ of the hierarchy we split the latent variable in two components, $z_i = (z_i^{\text{BU}}, z_i^{\text{TD}})$, which belong to a bottom-up (BU) and top-down (TD) inference path, respectively. More details on this will be given when introducing the inference network. The generative model of BIVA is illustrated in Figure 1a. We introduce a deterministic top-down path $d_{L-1}, \ldots, d_1$ that is parameterized with neural networks and receives as input at each layer i of the hierarchy the latent variable z_{i+1}. In the case of a convolutional model, this is done by concatenating $(z_{i+1}^{\text{BU}}, z_{i+1}^{\text{TD}})$ and d_{i+1} along the features' dimension. d_i can therefore be seen as a deterministic variable that summarizes all the relevant information coming from the stochastic variables higher in the hierarchy, $z_{>i}$. The latent variables z_i^{BU} and z_i^{TD} are conditioned on all the information in the higher layers, and are conditionally independent given $z_{>i}$. The joint distribution of the model is then given by:

$$p_\theta(x, \mathbf{z}) = p_\theta(x|\mathbf{z}) p_\theta(z_L) \prod_{i=1}^{L-1} p_\theta(z_i^{\text{BU}}|z_{>i}) p_\theta(z_i^{\text{TD}}|z_{>i}) \,,$$

where θ are the parameters of the generative model. The likelihood of the model $p_\theta(x|\mathbf{z})$ directly depends on z_1, and depends on $z_{>1}$ through the deterministic top-down path. Each stochastic latent variable $1, ..., L$ is parameterized by a Gaussian distribution with diagonal covariance, with one neural network $\mu(\cdot)$ for the mean and another neural network $\sigma(\cdot)$ for the variance. Since the z_{i+1}^{BU} and z_{i+1}^{TD} variables are on the same level in the generative model and of the same dimensionality, we share all the deterministic parameters going to the layer below. See Appendix B for details.

Bidirectional inference network. Due to the non-linearities in the neural networks that parameterize the generative model, the exact posterior distribution $p_\theta(\mathbf{z}|x)$ is intractable and needs to be approximated. As for VAEs, we therefore define a variational distribution, $q_\phi(\mathbf{z}|x)$, that needs to be flexible enough to approximate the true posterior distribution, as closely as possible. We define a bottom-up (BU) and a top-down (TD) inference path, which are computed sequentially when constructing the posterior approximation for each data point x, see Figure 1b. The variational distribution over the BU latent variables depends on the data x and on all BU variables lower in the hierarchy, i.e. $q_\phi(z_i^{\text{BU}}|x, z_{<i}^{\text{BU}})$, where ϕ denotes all the parameters of the BU path. z_i^{BU} has a direct dependency only on the BU variable below, z_{i-1}^{BU}. The dependency on $z_{<i-1}^{\text{BU}}$ is achieved, similarly to the generative model, through a deterministic bottom-up path $\widetilde{d}_1, \ldots, \widetilde{d}_{L-1}$.

The TD variables depend on the data and the BU variables lower in the hierarchy through the BU inference path, but also on all variables above in the hierarchy through the TD inference path, see Figure 1b. The variational approximation over the TD variables is thereby $q_{\phi,\theta}(z_i^{\text{TD}}|x, z_{<i}^{\text{BU}}, z_{>i}^{\text{BU}}, z_{>i}^{\text{TD}})$. Importantly, all the parameters of the TD path are shared with the generative model, and are therefore denoted as θ. The overall inference network can be factorized as follows:

$$q_\phi(\mathbf{z}|x) = q_\phi(z_L|x, z_{<L}^{\text{BU}}) \prod_{i=1}^{L-1} q_\phi(z_i^{\text{BU}}|x, z_{<i}^{\text{BU}}) q_{\phi,\theta}(z_i^{\text{TD}}|x, z_{<i}^{\text{BU}}, z_{>i}^{\text{BU}}, z_{>i}^{\text{TD}}) \, ,$$

where the variational distributions over the BU and TD latent variables are Gaussians whose mean and diagonal covariance are parameterized with neural networks that take as input the concatenation over the feature dimension of the conditioning variables. Training of BIVA is performed, as for VAEs, by maximizing the ELBO in eq. (1) with stochastic backpropagation and the reparameterization trick.

3.2 Motivation

BIVA can be seen as an extension of the LVAE in which we (i) add a deterministic top-down path and (ii) apply a bidirectional inference network. We will now provide the motivation and some intuitions on the role of these two novel components, that will then be empirically validated with the ablation study of Section 4.1.

Deterministic top-down path. Skip-connections represent one of the simplest yet most powerful advancements of deep learning in recent years. They allow constructing very deep neural networks, by better propagating the information throughout the model and reducing the issue of vanishing gradients. Skip connections form for example the backbone of deep neural networks such as ResNets [15], which have shown impressive performances on a wide range of classification tasks. Our goal in this paper is to build very deep latent variable models that are able to learn an expressive latent hierarchical representation of the data. In our experiments, we however found that the LVAE still had difficulties in activating the top latent variables for deeper hierarchies. To limit this issue, we add skip connections among the latent variables in the generative model by adding the deterministic top-down path, that makes each variable depend on all the variables above in the hierarchy (see Figure 1a for a graphical representation). This allows a better flow of information in the model and thereby avoids the collapse of latent variables. A related idea was recently proposed by [7], that add skip connections among the neural network layers parameterizing a shallow VAE with a single latent variable.

Bidirectional inference. The inspiration for the bidirectional inference network of BIVA comes from the work on Auxiliary VAEs (AVAE) by [37, 31]. An AVAE can be viewed as a shallow VAE with a single latent variable z and an auxiliary variable a that increases the expressiveness of the variational approximation $q_\phi(z|x) = \int q_\phi(z|a,x) q_\phi(a|x) \mathrm{d}a$. By making the inference network $q_\phi(z|a,x)$ depend on the stochastic variable a, the AVAE adds covariance structure to the posterior approximation over the stochastic unit z, since it no longer factorizes over its components $z^{(k)}$, i.e. $q_\phi(z|x) \neq \prod_k q_\phi(z^{(k)}|x)$. As discussed in the following, by factorizing the latent variables at each level of the hierarchy of BIVA we are able to achieve similar results without introducing additional auxiliary variables in the model. To see this, we can focus for example on the highest latent variable z_L. In BIVA, the presence of the z_i^{BU} variables makes the bottom-up inference path *stochastic*, as opposed to the deterministic BU path of the LVAE. While the conditional distribution $q_\phi(z_L|x, z_{<L}^{\text{BU}})$ still factorizes over the components of z_L, due to the stochastic BU variables the marginal distribution over z_L no longer factorizes, i.e. $q_\phi(z_L|x) = \int q_\phi(z_L|x, z_{<L}^{\text{BU}}) q_\phi(z_{<L}^{\text{BU}}|x) \mathrm{d}z_{<L}^{\text{BU}} \neq \prod_{k=1}^{K} q(z_L^{(k)}|x)$. Therefore, the BU inference path enables the learning of a complex covariance structure in the higher TD stochastic latent variables, which is fundamental in the model to extract *good* high-level semantic features from the data distribution. Notice that, in BIVA, only z_1^{BU} will have a marginally factorizing inference network.

3.3 Anomaly detection with BIVA

Anomaly detection is considered to be one of the most important applications of explicit density models. However, recent empirical results suggest that these models are not able to distinguish between two clearly distinctive data distributions [34], as they can assign a higher likelihood to data points from a data distribution that is very different from the one the model was trained on. Based on a thorough study, [34] states that the main issue is the fact that explicit density models tend to capture low-level statistics, as opposed to the high-level semantics that are preferable when doing anomaly detection. We hypothesize that the latent representations in the higher layers of BIVA can capture the high-level semantics of the data and that these can be used for improved anomaly detection.

In the standard ELBO from eq. (1), the main contribution to the expected log-likelihood term is coming from averaging over the variational distribution of the lower level latent variables. This will thus emphasize low-level statistics. So in order to perform anomaly detection with BIVA we instead need to emphasize the contribution from the higher layers. We can achieve this introducing an alternative score function inspired by the ELBO that partly replaces the inference network with the generative model, and uses therefore the generative hierarchy of the stochastic variables. In the following we define the hierarchy of stochastic latent variables as $\mathbf{z} = z_1, z_2, z_3, ..., z_L$ with $z_i = (z_i^{\text{BU}}, z_i^{\text{TD}})$. Instead of using as in the standard ELBO the variational approximation $q_\phi(\mathbf{z}|x)$ over all stochastic variables in the model, we use the prior distribution for the first k layers and the variational approximation from the inference network for the others, i.e. $p_\theta(z_{\leq k}|z_{>k}) q_\phi(z_{>k}|x, z_{\leq k}^{\text{BU}})$. In the computation of $q_\phi(z_{>k}|x, z_{\leq k}^{\text{BU}})$ we use samples $z_{\leq k}^{\text{BU}}$ from the inference network. Using this alternative distribution instead of $q_\phi(\mathbf{z}|x)$ in the ELBO in eq. (1), we define the score function for anomaly detection as:

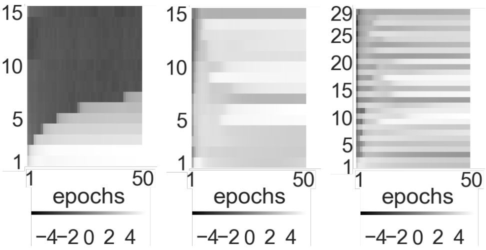
$$\mathcal{L}^{>k} = \mathbb{E}_{p_\theta(z_{\leq k}|z_{>k}) q_\phi(z_{>k}|x, z_{\leq k}^{\text{BU}})} \left[\log \frac{p_\theta(x|\mathbf{z}) p_\theta(z_{>k})}{q_\phi(z_{>k}|x, z_{\leq k}^{\text{BU}})} \right] . \tag{2}$$

$\mathcal{L}^{>0} = \mathcal{L}$ is the ELBO in eq. (1). As for the ELBO, we approximate the computation of $\mathcal{L}^{>k}$ with Monte Carlo integration. Sampling from $p_\theta(z_{\leq k}|z_{>k}) q_\phi(z_{>k}|x, z_{\leq k}^{\text{BU}})$ can be easily performed by obtaining samples $\widehat{z}_{>k}$ from the inference network, that are then used to sample $\widehat{z}_{\leq k}$ from the conditional prior $p_\theta(z_{\leq k}|\widehat{z}_{>k})$.

$\mathcal{L}^{>k}$ with higher values of k represents a useful metric for anomaly detection, as shown empirically in the experiments of Section 4.4. By only sampling the top $L - k$ variables from the variational approximation, in fact, we are forcing the model to only rely on the high-level semantics encoded in the highest variables of the hierarchy when evaluating this metric, and not on the low-level statistics encoded in the lower variables.

4 Experiments

BIVA is empirically evaluated by (i) an ablation study analyzing each novel component, (ii) likelihood and semi-supervised classification results on binary images, (iii) likelihood results on natural images, and (iv) an analysis of anomaly detection in complex data distributions. We employ a *free bits* strategy with $\lambda = 2$ [23] for all experiments to avoid latent variable collapse during the initial training epochs. Trained models are reported with 1 importance weighted sample, $\mathcal{L}_1$, and 1000 importance weighted samples, $\mathcal{L}_{1e3}$ [3]. We evaluate the natural image experiments by bits per dimension (bits/dim), $\mathcal{L}/(hwc \log(2))$, where h, w, c denote the height, width, and channels respectively. For a detailed



(a) LVAE $L = 15$ (b) LVAE+ $L = 15$ (c) BIVA $L = 15$

Figure 2: The $\log KL(q||p)$ for each stochastic latent variable as a function of the training epochs on CIFAR-10. (a) is a $L = N = 15$ stochastic latent layer LVAE with no skip-connections and no bottom-up inference. (b) is a $L = N = 15$ LVAE+ with skip-connections and no bottom-up inference. (c) is a $L = 15$ stochastic latent layer ($N = 29$ latent variables) BIVA for which $1, 2, ..., N$ denotes the stochastic latent variables following the order $z_1^{\text{BU}}, z_1^{\text{TD}}, z_2^{\text{BU}}, z_2^{\text{TD}}, ..., z_L$.

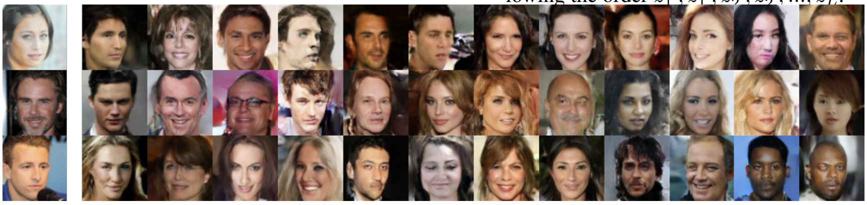


Figure 3: (left) images from the CelebA dataset preprocessed to 64x64 following [27]. (right) $\mathcal{N}(0, I)$ generations of BIVA with $L = 20$ layers that achieves a $\mathcal{L}_1 = 2.48$ bits/dim on the test set.

description of the experimental setup see Appendix C and the source code[1][2]. In Appendix D we test BIVA on complex 2d densities, while Appendix E presents initial results for the model on text.

4.1 Ablation Study

BIVA can be viewed as an extension of the LVAE from [50] where we add (i) extra dependencies in the generative model ($p_\theta(x|z_1) \rightarrow p_\theta(x|\mathbf{z})$ and $p_\theta(z_i|z_{i+1}) \rightarrow p_\theta(z_i|z_{>i})$) through the skip connections obtained with the deterministic top-down path and (ii) a bottom-up (BU) path of stochastic latent variables to the inference model. In order to evaluate the effects of each added component we define an LVAE with the exact same architecture as BIVA, but without the BU variables and the deterministic top-down path. Next, we define the LVAE+, where we add to the LVAE's generative model the deterministic top-down path. It is therefore the same model as in Figure 1 but without the BU variables. Finally, we investigate a LVAE+ model with $2L - 1$ stochastic layers. This corresponds to the depth of the hierarchy of the BIVA inference model $x \rightarrow z_1^{\text{BU}} \rightarrow \cdots \rightarrow z_{L-1}^{\text{BU}} \rightarrow z_L \rightarrow z_{L-1}^{\text{TD}} \rightarrow \cdots \rightarrow z_1^{\text{TD}}$. If this model is competitive with BIVA then it is an indication that it is the depth that determines the performance. The ablation study is conducted on the CIFAR-10 dataset against the best reported BIVA with $L = 15$ layers (Section 4.3), which means $2L - 1 = 29$ stochastic latent layers in the deep LVAE+.

Table 1 presents a comparison of the different model architectures. The positive effect of adding the skip connections in the generative models can be evaluated from the difference between the LVAE $L = 15$ and LVAE+ $L = 15$ results, for which there is close to a 0.2 bits/dim difference in the ELBO. Thanks to the more expressive posterior approximation obtained using its bidirectional inference network, BIVA improves the ELBO significantly w.r.t the LVAE+, by more than 0.3 bits/dim. Notice that a deeper hierarchy of stochastic latent variables in the LVAE+ will not necessarily provide a better likelihood performance, since the LVAE+ $L = 29$ performs worse than the LVAE+ $L = 15$ despite having significantly more parameters. In Figure 2 we plot for LVAE, LVAE+ and BIVA the KL divergence between the variational approximation over each latent variable

Table 1: A comparison of the LVAE with no skip-connections and no bottom-up inference, the LVAE+ with skip-connections and no bottom-up inference, and BIVA. All models are trained on the CIFAR-10 dataset.

	PARAM.	BITS/DIM
LVAE L=15, $\mathcal{L}_1$	10.85M	≤ 3.60
LVAE+ L=15, $\mathcal{L}_1$	11.36M	≤ 3.41
LVAE+ L=29, $\mathcal{L}_1$	21.99M	≤ 3.45
BIVA L=15, $\mathcal{L}_1$	18.48M	≤ 3.12

---

[1] Source code (Tensorflow): https://github.com/larsmaaloee/BIVA.

[2] Source code (PyTorch): https://github.com/vlievin/biva-pytorch.

Table 2: Test log-likelihood on statically binarized MNIST for different number of importance weighted samples. The finetuned models are trained for an additional number of epochs with no *free bits*, $\lambda = 0$. For testing resiliency we trained 4 models and evaluated the standard deviations to be ± 0.031 for $\mathcal{L}_1$.

	$-\log p(x)$
With autoregressive components	
PixelCNN [57]	$= 81.30$
DRAW [13]	< 80.97
IAFVAE [23]	≤ 79.88
PixelVAE [14]	≤ 79.66
PixelRNN [57]	$= 79.20$
VLAE [5]	≤ 79.03
Without autoregressive components	
Discrete VAE [42]	≤ 81.01
BIVA, $\mathcal{L}_1$	≤ 81.20
BIVA, $\mathcal{L}_{1e3}$	≤ 78.67
BIVA finetuned, $\mathcal{L}_1$	≤ 80.47
BIVA finetuned, $\mathcal{L}_{1e3}$	≤ 78.59

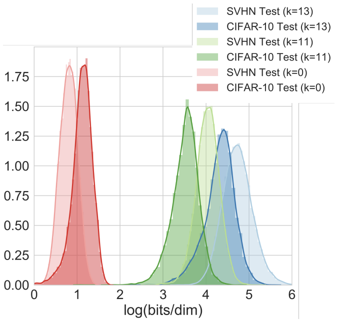


Figure 4: Histograms and kernel density estimation of the $\mathcal{L}^{>k}$ for $k = 13, 11, 0$ evaluated in bits/dim by a model trained on the CIFAR-10 train dataset and evaluated on the CIFAR-10 and the SVHN test set.

Table 3: Semi-supervised test error for BIVA on MNIST for 100 randomly chosen and evenly distributed labelled samples.

	Error %
M1+M2 [22]	3.33% (±0.14)
VAT [32]	2.12%
CatGAN [51]	1.91% (±0.10)
SDGM [31]	1.32% (±0.07)
LadderNet [38]	1.06% (±0.37)
ADGM [31]	0.96% (±0.02)
ImpGAN [44]	0.93% (±0.07)
TripleGAN [29]	0.91% (±0.58)
SSLGAN [6]	0.80% (±0.10)
BIVA	0.83% (±0.02)

Table 4: Test log-likelihood on CIFAR-10 for different number of importance weighted samples. We evaluated two different BIVA with various number of layers (L). For testing resiliency we trained 3 models and evaluated the standard deviations to be ± 0.013 for $\mathcal{L}_1$ and $L = 15$.

	bits/dim
With autoregressive components	
ConvDRAW [12]	< 3.58
IAFVAE $\mathcal{L}_1$ [23]	≤ 3.15
IAFVAE $\mathcal{L}_{1e3}$ [23]	≤ 3.12
GatedPixelCNN [56]	$= 3.03$
PixelRNN [57]	$= 3.00$
VLAE [5]	≤ 2.95
PixelCNN++ [45]	$= 2.92$
Without autoregressive components	
NICE [8]	$= 4.48$
DeepGMMs [58]	$= 4.00$
RealNVP [9]	$= 3.49$
DiscreteVAE++ [54]	≤ 3.38
GLOW [21]	$= 3.35$
Flow++ [16]	$= 3.08$
BIVA L=10, $\mathcal{L}_1$	≤ 3.17
BIVA L=15, $\mathcal{L}_1$	≤ 3.12
BIVA L=15, $\mathcal{L}_{1e3}$	≤ 3.08

and its prior distribution, $KL(q||p)$. This KL divergence is 0 when the two distributions match, in which case we say that the variable has collapsed, since its posterior approximation is not using any data-dependent information. We can see that while the LVAE is only able to utilize its lowest 7 stochastic variables, all variables in both LVAE+ and BIVA are active. We attribute this tendency to the deterministic top-down path that is present in both models, which creates skip-connections between all latent variables that allow to better propagate the information throughout the model.

4.2 Binary Images

We evaluate BIVA $L = 6$ in terms of test log-likelihood on statically binarized MNIST [43], dynamically binarized MNIST [28] and dynamically binarized OMNIGLOT [25]. The model parameterization and optimization parameters have been kept identical for all binary image experiments (see Appendix C). For each experiment on binary image datasets, we *finetune* each model by setting the free bits to $\lambda = 0$ until convergence in order to test the tightness of the $\mathcal{L}_1$ ELBO.

To the best of our knowledge, BIVA achieves state-of-the-art results on statically binarized MNIST, outperforming other latent variable models, autoregressive models, and flow-based models (see Table 2). Finetuning the model with $\lambda = 0$ improves the $\mathcal{L}_1$ ELBO significantly and achieves slightly better performance for the 1000 importance weighted samples. For dynamically binarized MNIST

	$\mathcal{L}^{>L-2}$	$\mathcal{L}^{>L-4}$	$\mathcal{L}^{>L-6}$	$\mathcal{L}^{>0}$
Model trained on CIFAR-10:				
CIFAR-10	79.36	35.34	20.93	3.12
SVHN	121.04	58.82	26.76	2.28
Model trained on FashionMNIST:				
FashionMNIST	228.38	107.07	-	94.05
MNIST	295.95	130.39	-	128.60

Table 5: The test $\mathcal{L}^{>k}$ for different values of k and train/test dataset combinations evaluated in bits/dim for natural images and negative log-likelihood for binary images (lower is better).

and OMNIGLOT, BIVA achieves similar improvements with $\mathcal{L}_{1e3} = 78.41$ (state-of-the-art) and $\mathcal{L}_{1e3} = 91.34$ respectively, see Tables 10 and 11 in Appendix G.

Semi-supervised learning. BIVA can be easily extended for semi-supervised classification by adding a categorical variable y to represent the class, as done in [22]. We add a classification model $q_\phi(y|x, z_{<L}^{\mathrm{BU}})$ to the inference network, and a class-conditional distribution $p_\theta(x|\mathbf{z}, y)$ to the generative model (see Appendix F for a detailed description). We train 5 different semi-supervised models on MNIST, each using a different set of just 100 randomly chosen and evenly distributed MNIST labels. Table 3 presents the classification results on the test set (mean and standard deviation over the 5 runs), that shows that BIVA achieves comparable performance to recent state-of-the-art results by generative adversarial networks.

4.3 Natural Images

We trained and evaluated BIVA $L = 15$ on 32x32 CIFAR-10, 32x32 ImageNet [57], and another BIVA $L = 20$ on 64x64 CelebA [27]. For the output decoding, we employ the discretized logistic mixture likelihood from [45] (see Appendix C for more details). In Table 4 we see that for the CIFAR-10 dataset BIVA outperforms other state-of-the-art non-autoregressive models and performs slightly worse than state-of-the-art autoregressive models. Notice that BIVA has fewer parameters (18.48M) than PixelCNN++ (28.57M parameters, [45]). For the 32x32 ImageNet dataset BIVA achieves better performance than flow-based models, but the performance gap to the autoregressive models remains large (Table 13 in Appendix G). This may be due to the added complexity (more categories) of the 32x32 ImageNet dataset, requiring an even more flexible model. More research should be invested in defining an improved architecture for BIVA that holds more parameters and thereby achieves better performances.

Figure 3 shows generated samples from the $\mathcal{N}(0, I)$ prior of a BIVA $L = 20$ trained on the CelebA dataset. From a visual inspection, the samples are far superior to previous natural image generations by latent variable models. We believe that previous claims stating that this type of model can only generate *blurry* images should be disregarded [27]. Rather the limited expressiveness/flexibility of previous models should be blamed. Additional samples from BIVA can be found in Appendix G.

4.4 Does BIVA know what it doesn't know?

We test the anomaly detection capabilities of BIVA replicating the most challenging experiments of [34]. We train BIVA $L = 15$ on the CIFAR-10 dataset, and evaluate eq. (2) for various values of k on the CIFAR-10 test set, the SVHN dataset [35] and the CelebA dataset. The results can be found in Table 5 and Figure 4, and are reported in terms of bits per dimension (lower is better). We see that for $k = 0$, corresponding to the standard ELBO, BIVA wrongly assigns lower values to data points from SVHN. This is in line with the results obtained with other explicit density models in [34], and shows that by using the standard ELBO the low-level image statistics prevail and the model is not able to correctly detect out-of-distribution samples. However, for higher values of k, the situation is reversed. We take this as an indication that BIVA uses the high-level semantics inferred from the data to better differentiate between the CIFAR-10 and the SVHN/CelebA distributions. We repeat the experiment training BIVA $L = 6$ on the FashionMNIST dataset (Table 5), and testing on the FashionMNIST test set and the MNIST dataset. Unlike the flow-based models used in [34], BIVA is able to learn a data distribution that can be used to detect anomalies with the standard ELBO (but also $k > 0$).

5 Conclusion

In this paper, we have introduced BIVA, that significantly improves performances over previously introduced probabilistic latent variable models and flow-based models. BIVA is able to generate natural images that are both sharp and coherent, to improve on semi-supervised classification benchmarks and, contrarily to other models, allows for anomaly detection using the extracted high-level semantics of the data.

References

[1] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8), 2013.

[2] S. Bowman, L. Vilnis, O. Vinyals, A. Dai, R. Jozefowicz, and S. Bengio. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*, 2015.

[3] Y. Burda, R. Grosse, and R. Salakhutdinov. Accurate and conservative estimates of mrf log-likelihood using reverse annealing. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2015.

[4] Y. Burda, R. Grosse, and R. Salakhutdinov. Importance Weighted Autoencoders. *arXiv preprint arXiv:1509.00519*, 2015.

[5] X. Chen, D. P. Kingma, T. Salimans, Y. Duan, P. Dhariwal, J. Schulman, I. Sutskever, and P. Abbeel. Variational Lossy Autoencoder. In *International Conference on Learning Representations*, 2017.

[6] Z. Dai, Z. Yang, F. Yang, W. W. Cohen, and R. Salakhutdinov. Good semi-supervised learning that requires a bad GAN. In *Advances in Neural Information Processing Systems*, 2017.

[7] A. B. Dieng, Y. Kim, A. M. Rush, and D. M. Blei. Avoiding latent variable collapse with generative skip models. *arXiv preprint arXiv:1807.04863*, 2018.

[8] L. Dinh, D. Krueger, and Y. Bengio. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*, 2014.

[9] L. Dinh, J. Sohl-Dickstein, and S. Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

[10] A. Dosovitskiy and T. Brox. Generating images with perceptual similarity metrics based on deep networks. In *Advances in Neural Information Processing Systems*, 2016.

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*. 2014.

[12] K. Gregor, R. D. J. Besse, Fredric, I. Danihelka, and D. Wierstra. Towards conceptual compression. *arXiv preprint arXiv:1604.08772*, 2016.

[13] K. Gregor, I. Danihelka, A. Graves, and D. Wierstra. Draw: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015.

[14] I. Gulrajani, K. Kumar, F. Ahmed, A. Ali Taiga, F. Visin, D. Vazquez, and A. Courville. PixelVAE: A latent variable model for natural images. *arXiv e-prints*, 1611.05013, Nov. 2016.

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

[16] J. Ho, X. Chen, A. Srinivas, Y. Duan, and P. Abbeel. Flow++: Improving flow-based generative models with variational dequantization and architecture design. *arXiv preprint arXiv:1902.00275*, 2019.

[17] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[18] M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 2013.

[19] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, Nov. 1999.

[20] D. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*, 12 2014.

[21] D. P. Kingma and P. Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems*, 2018.

[22] D. P. Kingma, D. J. Rezende, S. Mohamed, and M. Welling. Semi-Supervised Learning with Deep Generative Models. In *Proceedings of the International Conference on Machine Learning*, 2014.

[23] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling. Improved variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems*. 2016.

[24] M. Kingma, Diederik P; Welling. Auto-Encoding Variational Bayes. *arXiv preprint arXiv:1312.6114*, 12 2013.

[25] B. M. Lake, R. R. Salakhutdinov, and J. Tenenbaum. One-shot learning by inverting a compositional causal process. In *Advances in Neural Information Processing Systems*. 2013.

[26] H. Larochelle and I. Murray. The neural autoregressive distribution estimator. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, 2011.

[27] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther. Autoencoding beyond pixels using a learned similarity metric. In *Proceedings of the International Conference on Machine Learning*, 2016.

[28] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2278–2324, 1998.

[29] C. Li, K. Xu, J. Zhu, and B. Zhang. Triple generative adversarial nets. *arXiv preprint arXiv:1703.02291*, 2017.

[30] L. Maaløe, M. Fraccaro, and O. Winther. Semi-supervised generation with cluster-aware generative models. *arXiv preprint arXiv:1704.00637*, 2017.

[31] L. Maaløe, C. K. Sønderby, S. K. Sønderby, and O. Winther. Auxiliary Deep Generative Models. In *Proceedings of the International Conference on Machine Learning*, 2016.

[32] T. Miyato, S.-i. Maeda, M. Koyama, K. Nakae, and S. Ishii. Distributional Smoothing with Virtual Adversarial Training. *arXiv preprint arXiv:1507.00677*, 7 2015.

[33] A. Mnih and K. Gregor. Neural variational inference and learning in belief networks. In *Proceedings of the International Conference on Machine Learning*, pages 1791–1799, 2014.

[34] E. Nalisnick, A. Matsukawa, Y. W. Teh, D. Gorur, and B. Lakshminarayanan. Do deep generative models know what they don't know? *arXiv preprint arXiv:1810.09136*, 2018.

[35] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. In *Deep Learning and Unsupervised Feature Learning, workshop at Neural Information Processing Systems 2011*, 2011.

[36] J. Paisley, D. M. Blei, and M. I. Jordan. Variational bayesian inference with stochastic search. In *Proceedings of the International Conference on Machine Learning*, pages 1363–1370, 2012.

[37] R. Ranganath, D. Tran, and D. M. Blei. Hierarchical variational models. In *Proceedings of the International Conference on Machine Learning*, 2016.

[38] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko. Semi-supervised learning with ladder networks. In *Advances in Neural Information Processing Systems*, 2015.

[39] D. J. Rezende and S. Mohamed. Variational Inference with Normalizing Flows. In *Proceedings of the International Conference on Machine Learning*, 2015.

[40] D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. *arXiv preprint arXiv:1401.4082*, 04 2014.

[41] D. J. Rezende and F. Viola. Taming vaes. *arXiv preprint arXiv:1810.00597*, 2018.

[42] J. T. Rolfe. Discrete variational autoencoders. In *Proceedings of the International Conference on Learning Representations*, 2017.

[43] R. Salakhutdinov and I. Murray. On the quantitative analysis of deep belief networks. In *Proceedings of the International Conference on Machine Learning*, 2008.

[44] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. *arXiv preprint arXiv:1606.03498*, 2016.

[45] T. Salimans, A. Karparthy, X. Chen, and D. P. Kingma. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *arXiv preprint:1701.05517, 2017*, 2017.

[46] T. Salimans and D. P. Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Proceedings of the International Conference on Neural Information Processing Systems*, 2016.

[47] T. Salimans, D. P. Kingma, and M. Welling. Markov chain monte carlo and variational inference: Bridging the gap. In *Proceedings of the International Conference on Machine Learning*, 2015.

[48] S. Semeniuta, A. Severyn, and E. Barth. A hybrid convolutional variational autoencoder for text generation. *arXiv preprint arXiv:1702.02390*, 2017.

[49] H. Shah, B. Zheng, and D. Barber. Generating sentences using a dynamic canvas, 2018.

[50] C. K. Sønderby, T. Raiko, L. Maaløe, S. K. Sønderby, and O. Winther. Ladder variational autoencoders. In *Advances in Neural Information Processing Systems 29*. 2016.

[51] J. Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. *arXiv preprint arXiv:1511.06390*, 2015.

[52] J. M. Tomczak and M. Welling. Improving variational auto-encoders using householder flow. *arXiv preprint arXiv:1611.09630*, 2016.

[53] D. Tran, R. Ranganath, and D. M. Blei. Variational Gaussian process. In *Proceedings of the International Conference on Learning Representations*, 2016.

[54] A. Vahdat, W. G. Macready, Z. Bian, A. Khoshaman, and E. Andriyash. DVAE++: discrete variational autoencoders with overlapping transformations. In *Proceedings of the International Conference on Machine Learning*, 2018.

[55] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[56] A. van den Oord, N. Kalchbrenner, O. Vinyals, L. Espeholt, A. Graves, and K. Kavukcuoglu. Conditional image generation with pixelcnn decoders. *arXiv preprint arXiv:1606.05328*, 2016.

[57] A. van den Oord, K. Nal, and K. Kavukcuoglu. Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759*, 01 2016.

[58] A. van den Oord and B. Schrauwen. Factoring variations in natural images with deep gaussian mixture models. In *Advances in Neural Information Processing Systems*, 2014.

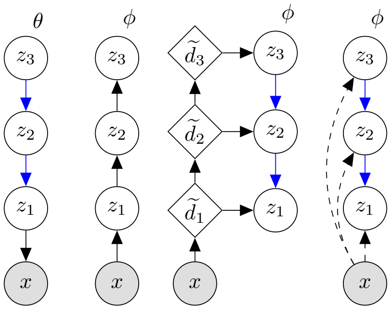
[59] S. Zhao, J. Song, and S. Ermon. Towards deeper understanding of variational autoencoding models. *arXiv preprint arXiv:1702.08658*, 2017.

[60] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27, 2015.

A Deep Learning and Variational Inference

The introduction of stochastic backpropagation [36, 18] and the variational auto-encoder (VAE) [24, 40] has made approximate Bayesian inference and probabilistic latent variable models applicable to machine learning problems considering complex data distributions, e.g. natural images, audio, and text. The VAE is a generative model parameterized by a neural network θ and is defined by an observed variable x that depends on a hierarchy of stochastic latent variables $\mathbf{z} = z_1, ..., z_L$ so that: $p_\theta(x, \mathbf{z}) = p_\theta(x|z_1)p_\theta(z_L)\prod_{i=1}^{L-1} p_\theta(z_i|z_{i+1})$. This is illustrated in Figure 5a.

The distributions $p_\theta(z_i|z_{i+1})$ over the latent variables of the VAE are normally defined as Gaussians with diagonal covariance, whose parameters depend on the previous latent variable in the hierarchy (with the top latent variable $p_\theta(z_L) = \mathcal{N}(z_L; 0, I)$). The likelihood $p_\theta(x|z_1)$ is typically a Gaussian distribution for continuous data, or a Bernoulli distribution for binary data.



(a) (b) (c) (d)

Figure 5: (a) Generative model of a VAE/LVAE with $L = 3$ stochastic variables, (b) VAE inference model, (c) LVAE inference model, and (d) skip connections among stochastic variables in the LVAE where dashed lines denote a skip-connection. Blue arrows indicate that there are shared parameters between the inference and generative model.

In order to learn the parameters θ we seek to maximize the log marginal likelihood over a training set: $\sum_i \log p_\theta(x_i) = \sum_i \log \int p_\theta(x_i, \mathbf{z}_i)d\mathbf{z}_i$. However, complex data distributions require an expressive model, which makes the above integral intractable. In order to circumvent this, we use Variational Inference [19] and introduce a posterior approximation $q_\phi(\mathbf{z}|x)$, known as *inference network* or *encoder*, that is parameterized by a neural network ϕ. Using Jensen's inequality we can derive the *evidence lower bound* (ELBO), a lower bound to the integral in the marginal likelihood which is a function of the variational approximation $q_\phi(\mathbf{z}|x)$ and the generative model $p_\theta(x, \mathbf{z})$:

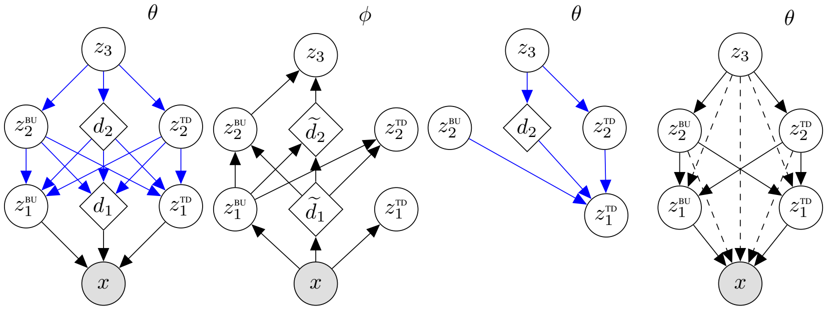
$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(\mathbf{z}|x)}\left[\log \frac{p_\theta(x, \mathbf{z})}{q_\phi(\mathbf{z}|x)}\right] \equiv \mathcal{L}(\theta, \phi) \ . \tag{3}$$

The parameters θ and ϕ can be optimized by maximizing the ELBO with stochastic backpropagation and the reparameterization trick, which allows using gradient ascent algorithms with low variance gradient estimators [24, 40]. As illustrated in Figure 5b, in a VAE the variational approximation is factorized with a bottom-up structure, $q_\phi(\mathbf{z}|x) = q_\phi(z_1|x)\prod_{i=1}^{L-1} q_\phi(z_{i+1}|z_i)$, so that each latent variable is conditioned on the variable below in the hierarchy. For ease of computation, all the factors in the variational approximation are typically assumed to be Gaussians whose mean and diagonal covariance are parameterized by neural networks.

Latent variable collapse in VAEs. A deep hierarchy of latent stochastic variables will result in a more expressive model. However, the additional variables come at a price. As shown in [5, 30], we can rewrite the ELBO (eq. (1)):

$$\mathcal{L}(\theta, \phi) = \mathbb{E}_{q_\phi(\mathbf{z}|x)}\left[\log \frac{p_\theta(x, z_{<L}|z_L)}{q_\phi(z_{<L}|x)}\right] - \mathbb{E}_{q_\phi(z_{<L}|x)}\left[KL[q_\phi(z_L|z_{<L})||p_\theta(z_L))]\right] \ .$$

From the above, it becomes obvious that, during the optimization of the VAE, the top stochastic latent variables may have a tendency to *collapse* into the prior, i.e. $q_\phi(z_L|z_{<L}) = p_\theta(z_L) = \mathcal{N}(z_L; 0, I)$, if the model $p_\theta(x, z_{<L}|z_L)$ is powerful enough. This is supported by empirical results in [50, 2] amongst others. The tendency has limited the applicability of deep VAEs in problems with complex data distributions, and has pushed VAE research towards the extension of shallow VAEs with autoregressive models, that allow capturing a *lossy* representation in the latent space while achieving strong generative performances [14, 5]. Another research direction has focused on learning more complex prior distributions through normalizing flows [39, 52, 23]. Our research considers instead the original goal of building expressive models that can exploit a deeper hierarchy of stochastic latent variables while avoiding variable collapse.



(a) Generative model (b) BU inference (c) TD inference (d) Variable dependency

Figure 6: A $L = 3$ layered BIVA with (a) the generative model, (b) bottom-up (BU) inference path, (c) top-down (TD) inference path, and (d) variable dependency of the generative models where dashed lines denote a skip-connection. Blue arrows indicate that the deterministic parameters are shared within the generative model or between the generative and inference model.

B Detailed Model Description

Generative model. The generative model (see Figure 6a) has a top-down path going from z_L through the intermediary stochastic latent variables to x. Between each stochastic layer there is a ResNet block with M layers set up similarly to [45]. Weight normalization [46] is applied in all neural network layers. In the generative model, the BU and TD units are not distinguished so we write $z_i = (z_i^{\mathrm{BU}}, z_i^{\mathrm{TD}})$. We use $f_{i,j}$ to denote the neural network function (a function of generative model parameters θ) of ResNet layer j associated with stochastic layer i. The feature maps are written as $d_{i,j}$. The generative process can then be iterated as $z_L \sim \mathcal{N}(0, I)$ and $i = L-1, L-2, \ldots, 1$:

$$
\begin{aligned}
d_{i,0} &= z_{i+1} && (4)\\
d_{i,j} &=< f_{\theta_{i,j}}(d_{i,j-1}); d_{i+1,j} > \ \textbf{for}\ j = 1, ..., M && (5)\\
z_i &= \mu_{\theta,i}(d_{i,M}) + \sigma_{\theta,i}(d_{i,M}) \otimes \epsilon_i \ , && (6)
\end{aligned}
$$

where $d_{L,j} = 0$, $<;>$ denotes concatenation of feature maps in the convolutional network and hidden units in the fully connected network, $\epsilon \sim \mathcal{N}(0, I)$ and $\mu(\cdot)$ and $\sigma(\cdot)$ are parameterized by neural networks. To complete the generative model $p(x|\mathbf{z})$ is written in terms of z_1 and d_1 through a ResNet block f_0.

Inference model. The inference model (see Figure 6b and 6c) consists of a bottom-up (BU) and top-down (TD) paths such that bottom-up stochastic units only receive bottom-up information whereas the top-down units receive both bottom-up and top-down information. The top-down path shares parameters with the generative model. For each stochastic latent variable z_i in $i = 1, ..., L$ we use a ResNet block with M layers and there are associated neural network functions $g_{i,j}, j = 1, \ldots, M$ with parameters collectively denoted by ϕ. The deterministic feature map of layer i, j is denoted by $\tilde{d}_{i,j}$:

$$
\begin{aligned}
\tilde{d}_{i,0} &= \begin{cases} x & i = 1 \\ < z_{i-1}; \tilde{d}_{i-1,M} > & \text{otherwise} \end{cases} && (7)\\
\tilde{d}_{i,j} &=< g_{i,j}(\tilde{d}_{i,j-1}); \tilde{d}_{i-1,j} > \ \textbf{for}\ j = 1, ..., M \ , && (8)\\
z_i^{\mathrm{BU}} &= \mu_i^{\mathrm{BU}}(\tilde{d}_{i,M}) + \sigma_i^{\mathrm{BU}}(\tilde{d}_{i,M}) \otimes \epsilon_i^{\mathrm{BU}} && (9)
\end{aligned}
$$

where $\epsilon \sim \mathcal{N}(0, I)$. Finally, to infer the top-down latent we use the bottom-up latent z_i^{TD} inferred in eq. (9) and pass them through the generative path eq. (5) for $i = L-1, L-2, \ldots, 2$ to determine $d_{i,M}$ and

$$
z_i^{\mathrm{TD}} = \mu_i^{\mathrm{TD}}(< \tilde{d}_{i,M}; d_{i,M} >) + \sigma_i^{\mathrm{TD}}(< \tilde{d}_{i,M}; d_{i,M} >) \otimes \epsilon_i^{\mathrm{TD}} \ . \qquad (10)
$$

C Experimental Setup

Throughout all experiments, we follow the BIVA model description that is described in detail in Appendix B and F.

Optimization. All models are optimized using Adamax [20] with a hyperparameter setting similar to the one used in [23]. They are trained with a batch-size of 48 where the binary image experiments are trained on a single GPU and the natural image experiments are trained on two GPUs (by splitting the batch in 2 and then taking the mean over the gradients). For evaluation, we use exponential moving averages of the parameters space, similar to [23, 45].

Binary image architecture. BIVA has $L = 6$ layers. The g_{ϕ_1} neural networks are defined by $M = 3$, 64x5x5 (number of kernels x kernel width x kernel height) convolutional layers and an overall stride of 2. Neural networks $i = 2, ..., 6$ are defined by four $M = 3$, 64x3x3 convolutional layers. The final neural network, $i = 6$, applies a stride of 2. All stochastic latent variables are densely connected layers of dimension $48, 40, 32, 24, 16, 8$ for $1, ..., L$ respectively. We apply a dropout rate of 0.5 for both the deterministic layers in the generative as well as the inference model.

Natural image architecture (32x32). BIVA has $L = 15$ layers. The g_{ϕ_1} neural networks are defined by $M = 3$, 96x5x5 convolutional layers and an overall stride of 2. Neural networks $i = 2, ..., 15$ are defined by $M = 3$, 96x3x3 convolutional layers. Neural networks 11 and 15 are defined with a stride of 2. All stochastic latent variables are parameterized by convolutional layers with $38, 36, 34, ..., 10$ feature maps for $1, 2, 3, ..., L$ respectively. The kernel width and height of the stochastic latent variables are defined similarly to the dimension of the subsequent output after striding. We apply a dropout rate of 0.2 in the deterministic layers of the inference model.

Natural image architecture (64x64). BIVA has $L = 20$ layers. The g_{ϕ_1} and g_{ϕ_2} neural networks are defined by $M = 3$, 64x7x7 and 64x5x5 convolutional layers respectively with a stride of 2 in each. Neural networks $i = 3, ..., 11$ are defined by $M = 3$ 64x3x3 convolutional layers. Neural network 11 is defined with a stride of 2. Neural networks $i = 12, ..., 20$ are defined by $M = 3$, 128x3x3 convolutional layers and network 20 has a stride of 2. All stochastic latent variables are parameterized by convolutional layers with $20, 19, 18, ..., 1$ feature maps for $1, 2, 3, ..., L$ respectively. The kernel width and height of the stochastic latent variables are defined similarly to the dimension of the subsequent output after striding. We apply a dropout rate of 0.2 in the deterministic layers of the inference model.

D Modeling Complex 2D Densities

	POTENTIAL $U(\mathbf{Z})$
1:	$\frac{1}{2}\left(\frac{\|\mathbf{z}\|-2}{0.4}\right)^2 - \ln\left(e^{-\frac{1}{2}\left[\frac{\mathbf{Z}_1-2}{0.6}\right]^2} + e^{-\frac{1}{2}\left[\frac{\mathbf{Z}_1+2}{0.6}\right]^2)}\right)$
2:	$\frac{1}{2}\left[\frac{\mathbf{Z}_2-w_1(\mathbf{Z})}{0.4}\right]^2$
3:	$-\ln\left(e^{-\frac{1}{2}}\left[\frac{\mathbf{Z}_2-\boldsymbol{w}_1(\mathbf{Z})}{0.35}\right]^2 + e^{-\frac{1}{2}\left[\frac{\mathbf{Z}_2-\boldsymbol{w}_1(\mathbf{Z})+w_2(\mathbf{Z})}{0.35}\right]^2)}\right)$
4:	$-\ln\left(e^{-\frac{1}{2}\left[\frac{\mathbf{Z}_2-w_1(\mathbf{Z})}{0.4}\right]^2} + e^{-\frac{1}{2}\left[\frac{\mathbf{Z}_2-w_1(\mathbf{Z})+w_3(\mathbf{Z})}{0.35}\right]^2)}\right)$
	WITH $w_1(\mathbf{z}) = \sin\left(\frac{2\pi\mathbf{z}_1}{4}\right)$, $w_2(\mathbf{z}) = 3e^{-\frac{1}{2}\left[\frac{(\mathbf{Z}_1-1)}{0.6}\right]^2}$, $w_3(\mathbf{z}) = 3\sigma\left(\frac{\mathbf{Z}_1-1}{0.3}\right)$ AND $\sigma(x) = 1/\left(1+e^{-x}\right)$.

Table 6: Potentials defining the target densities $p(\mathbf{z}) = \frac{e^{-U(\mathbf{z})}}{Z}$.

Problem. [31] showed that Variational Auto-Encoders can fit complex posterior distributions for the latent space using the inference model $q_\phi(z|x)$, parameterized as a fully factorized Gaussian and $p(x)$ being a simple diagonal Gaussian. In table 6, we define complex non-Gaussian densities using a potential model $U(\mathbf{Z})$, as described in [39]. While modeling such distributions remains within the reach of an adequately complex Variational Autoencoder, optimizing such a model remains challenging.

Objective. Similarly to [31], we choose $p(x)$ to be an isotropic Gaussian and we model the target density using the top stochastic variable: $p(z_L) = \frac{e^{U(z)}}{Z}$. This results in the following bound:

$$\log Z \geq \mathbb{E}_{q_\phi(x,\mathbf{z})}\left[U(z_L) + \log \frac{p_\theta(x|z_1)}{q_\phi(x)} + \sum_{i=1}^{L-1} \log \frac{p_\theta(z_i|z_{i+1})}{q_\phi(z_{i,TD}|z_{i+1},x)q_\phi(z_{i+1}|z_{i,BU},x)}\right] . \quad (11)$$

Experimental Setup. We test BIVA against the VAE and LVAE models using the same number of stochastic variables, hence the models use the same number of intermediate layers. All models are implemented using 5 stochastic layers, MLPs with one hidden layer of size 128 and with residual connections. The chosen architecture is voluntary kept minimal, therefore the task remains challenging for all models.

We train all models for $1e^4$ iterations using the Adamax optimizer. We use batch sizes of size 512. The potential is linearly annealed from 0.1 to 1 during $5e^3$ steps. In order to avoid posterior collapse, 0.5 *freebits* are applied to each stochastic layer. The learning rate is linearly increased from $1e^{-5}$ to $3e^{-3}$ and exponentially annealed back to $1e^{-5}$.

In order to measure the quality of the posterior density, we estimate $KL(q(z_L)||p(z_L))$ using $1e^6$ posterior samples evaluated using a grid of size $(-2,2)^2$ with a resolution of 100×100. Each model is trained 100 times for each density.

Results. According to the approximate $KL(q(z_L)||p(z_L))$, we found that BIVA tends to learn a posterior that lies closer to the target density. Figure 7 shows that BIVA often learns more complex features than the baseline models, which posteriors remain closer to the modes. Figure 7 reveals that LVAE is able to find solutions that are competitive with the best BIVA samples according to $KL(q(z_L)||p(z_L))$. However, this happens very rarely whereas BIVA has a more robust optimization behaviour.

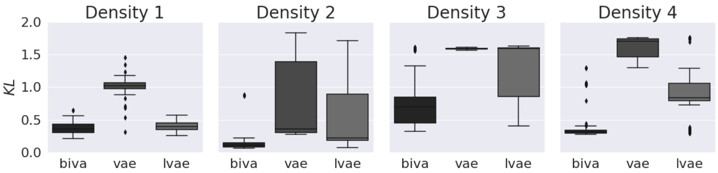


Figure 7: Distribution of the $KL(q(z_L)||p(z_L)))$ estimate for each model, each target density $p(z_L)$ and for different initial random seeds. We collected 100 runs for each model and for each density. We found that BIVA behaves more consistently and often yield better approximations than the baseline models.

E Initial Results on Text Generation Tasks

Optimizing generative models coupled with autoregressive models is a difficult task. Such coupling causes the posterior to collapse, and the latent variables are ignored. Nonetheless, autoregressive components remain a cornerstone of the generative models for text [2, 48, 49]. In order to enforce the model to use the latent variable, previous efforts aimed at weakening the decoder using powerful regularizing *tricks*, such as word dropout [2]. We investigate the use of BIVA in the context of sentence modeling without weakening the decoder. We show that it allows optimizing the latent variables more effectively, resulting in a higher measured KL when compared to the RNN-VAE [2] and the Hybrid VAE [48].

Dataset. We use the Bookcorpus dataset [60] of sentences of maximum 40 words, no preprocessing is performed and sentences are tokenized using the white spaces. We defined a vocabulary of 20000

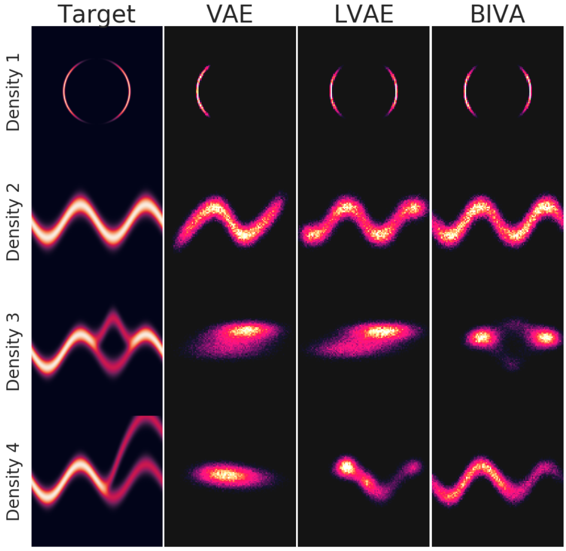


Figure 8: Target densities $p(z_L)$ and the median posterior distributions $q(z_L)$ for each model according to $KL(q(z_L)||p(z_L)))$ out of 100 runs for each model and for each density.

	PARAMETERS	$-\log p(x)$	KL	PPL
Results with autoregressive components, no dropout				
LSTM	$15.0M$	$= 41.49$	$-$	36.28
RNN-VAE [2], $\mathcal{L}_1$, WARMUP	$23.7M$	≤ 42.09	1.61	38.21
RNN-VAE [2], $\mathcal{L}_1$, FINETUNED	$23.7M$	≤ 42.41	5.13	39.26
HYBRID VAE [48], $\mathcal{L}_1$, FINETUNED	$23.7M$	≤ 42.24	4.67	38.70
BIVA L=7, $\mathcal{L}_1$, FINETUNED	$23.0M$	≤ 42.34	10.15	39.04
Results without autoregressive components, no dropout				
HYBRID VAE [48], $\mathcal{L}_1$, FINETUNED	$15.0M$	≤ 54.53	14.10	112.1
BIVA L=7 FINETUNED, $\mathcal{L}_1$	$14.0M$	≤ 54.13	15.33	108.3

Table 7: Test performances on the BookCorpus with 1 importance weighted sample (sentences limited to 40 words). The RNN-VAE and Hybrid VAE are are trained and evaluated from our own implementation.

words and filtered out the sentences that contain non-indexed tokens. We randomly sampled 10000 sentences for testing and used the remaining 56M sentences for training.

Models. We couple BIVA with an LSTM decoder, using the output of the convolutional model as an input sequence for the auto-regressive model. We compare our model against a LSTM language model [17], the RNN-VAE [2], and the Hybrid VAE [48], which couples a convolutional architecture with an LSTM decoder. We also perform experiments without using autoregressive components.

All LSTM models are parameterized by 1024 units and we use embeddings of dimension 512. This results in an RNN-VAE model with 23.7M parameters and we limit the other models to use the same total number of parameters. This results in using a limited number of stochastic layers for the BIVA and small a small number of kernels of 128.

Training. We trained the models for 5 epochs with an initial learning rate of $2e^{-3}$ using the Adamax optimizer. We used batches of size 512 and used only one stochastic sample. We train all latent variable models using the *freebits* method from [23] with an initial KL budget of 30 nats distributed equally over the stochastic variables and we incrementally decrease the *freebits* value *on plateau*. We also train the RNN-VAE baseline using the deterministic warmup method [2, 50] for comparison.

Likelihood and latent variables usage. We report the test set results in table 7 and test samples in 8 and reconstructions in table 9. While BIVA without the autoregressive decoder is not competitive with an LSTM language model, we observe that replacing the LSTM inference model by a BIVA model allows exploiting the latent space more actively, which results in a higher measured KL than the RNN-VAE and Hybrid VAE baselines.

BIVA+LSTM	RNN-VAE
he said .	“ two .
i tried to think of something to say to him , but he was already on his way back to the house .	“ you do n’t have to do this . ”
it sounded as if he was going to say something .	the light from the lamp was dim , but the light was dim and the room was dark .
“ and that ’s why you ’re coming . ”	or a nuclear bomb , or something .
“ what ? ”	“ the baby ? ”
she swallowed .	“ you ’re not going to kill me . ”
“ i want you . ”	she was n’t going to .
glancing up , i saw the way he was staring at me with a look of pure hatred .	“ i guess we could have been more careful , ” he said .
i need a favor . ”	there are some things that are not good .
he did n’t .	“ you ’re a good man .
you ’re not dead .	i had n’t been able to get it out .
i stood , and he followed .	“ you ’re going to have to be careful , ” he said .
“ can i sit on the couch and talk ? ”	it ’s not a bad idea .
“ it was n’t until i was fifteen , i was n’t in the mood to be around .	he asked .
i looked down at my lap .	“ this is a bad idea , ” he said , his voice a little hoarse .
the smile disappeared .	“ i ’m sure he ’s in love with you .
it was hard to tell which one was more of a rock .	as he stepped out of the car , he saw the man standing in the doorway , his eyes wide and his face pale .
i ’m not sure it ’s a good idea .	.
the first two .	“ no .
he was there .	“ in the meantime , i need to get some sleep , ” i said .
“ all of you , ” joe said .	i was n’t .
he did n’t care if he was n’t a vampire .	did i want to talk to you ?
her mouth curved up , then she nodded .	“ i want to hear you say it . ”
just tell me what you want in the end .	the train was already in the driveway .
and again .	“ good .
the other man ’s voice was hoarse and ragged .	i just needed to get out of here , and i needed to get out of here .
i had n’t known that was a bad idea , but i had n’t been able to get it out of my head .	“ this is a good idea .
your brother is the most important thing to me .	“ hey . ”
you dont need to go to the police , right ?	she took a deep breath and let it out .
there was a long silence .	then he kissed her .
i looked up .	i felt a warm hand on my shoulder and a warm smile spread across my face .
he nodded , and he looked at me , and i could tell he was thinking about it .	“ he ’s dead . ”
“ hang on , baby .	at the time , i was going to have to get out of the house .
we had to be close to the city , and we could n’t afford to be here .	he was so close to the edge of the bed .
you know , it would be better if you were n’t so stupid . ”	“ i do n’t know .
excuse me ?	“ i do n’t have a choice . ”
you know how much i love you , too .	i know i ’m not going to let him touch me , but i do .
a woman ’s voice , a voice that was familiar .	i could n’t see the face of the man who ’d just been in the doorway .
i have a very important business to attend to , and i ’m going to have to make a decision .	in the end , we all know that we are not going to be able to get out of this .
they sat on the small wooden table in the center of the room .	“ yes .
“ it ’s fine . ”	“ what are you doing here ? ”
she felt a rush of relief .	so the only thing that mattered was that he was here .
maria , he says .	neither of them spoke .
what ?	from now on , you will be able to get out of here .
“ it does n’t seem like a lot to me , ” he said .	the thought of having to kill him made him want to kill her .
he ’d told her everything .	the other two were staring at me , their eyes wide .
“ she ’s in shock .	i did n’t want to be a part of it , but i was n’t going to let it go .
“ after all , ” he murmured , “ i ’m going to go get the rest of the stuff . ”	“ i do n’t want to talk about it .
and then , finally , she ’d done it .	she looked at him , her eyes wide .
her words were a whisper , but it was n’t enough .	“ that ’s what you ’re going to do .

Table 8: Samples decoded from the prior of the BIVA with LSTM decoder and baseline RNN-VAE.

input	BIVA+LSTM	RNN-VAE
“ a sad song , being sung alone in the basement . ”	“ it sounds like you ’ve been through a lot . ”	“ you ’re going to be a great father . ”
more often , though , wherever she sank , beck was there .	more than anything , she wanted to be with him .	in the end , we all knew what was going on .
he looked just about as pale as i had ever seen him .	he ’s still a lot more than a friend .	he was n’t going to let her go .
caleb turned and shoved him back as he took his true form .	he lifted me up , his arms still wrapped around my waist .	he was standing in the doorway , his hands folded in front of him .
i gasped , tried to pull away , squeezed my legs together .	i gasped , and he was n’t able to stop himself .	i felt my body tense , and i could n’t help but smile .
i agreed as i adjusted myself and sat heavily in my chair .	i tried to ignore it , but my eyes were still closed .	i did n’t want to be the one to tell him .
you bind me , UNK in darkness , though , in light .	he ’d decided to take her home , to make her feel safe .	he was more than willing to let her go .
they promise me things , ask me questions , whisper and plead .	they might be able to do something about it , but they do n’t .	“ we need to talk , ” he said , his voice low .
i glowed as i held the bear , almost bigger than me .	i started to close my eyes , but he was too strong .	i could n’t help but smile at the sight of her .
i wonder how much he pays them to be his guard dogs .	i had to admit that it was n’t a good idea .	i do n’t want to be a part of this .
“ hmmm , ” richard muttered , and headed up the path .	“ jesus , ” he said , his voice barely audible .	“ but you ’re going to be a father .
he was happy that he had found it in the UNK hall .	he was n’t going to be the one to go .	he was n’t sure if he was going to make it .
at the shack , at the condo , at the hangar . ”	at the moment , the only thing that mattered was that he was n’t alone .	he was staring at the floor , his eyes wide .
“ i ’d pop to go to the dance with you . ”	“ i ’d prefer to go to the hospital .	“ i ’m going to go to the bathroom .
someday , i ’ll share them with the rest of the world .	and now i have a lot of my own .	“ we ’re going to have to do something about it .
“ maybe i ’m not the right person for this one ” .	“ maybe we can get a little more of a ride . ”	“ i do n’t think you ’re going to be able to do that . ”
“ gin is my sister , and she ’s coming with me .	“ there ’s a chance i can get a little more sleep . ”	“ if you want to , i ’ll be there . ”
thick desire stormed her ... along with a bittersweet curl of emotion .	the tension was gone , and he was n’t looking at me .	the air smelled of stale cigarette smoke .
they caused him to stagger back and drove him to the ground .	they had to be at the top of the hill .	he ’d found a way to get her to safety .
you ’re not much of a friar , friar , he says .	you ’re not supposed to be around here , are you ? ”	you ’re not going to be able to do that , are you ? ”

Table 9: Reconstruction of samples from the test set using the BIVA with LSTM decoder and the RNN-VAE baseline. The samples are decoded from the posterior distribution by using greedy decoding, without teacher forcing.

F Semi-Supervised Learning

When defining BIVA for semi-supervised classification tasks we follow the approach described for the M2 model in [22]. In addition to BIVA, described in detail in Appendix B, we introduce a classification model $q_\phi(y|x, z^{\text{BU}}_{<L})$ in the inference model, where y is the class variable, and a Categorical latent variable dependency in the generative model.

Inference model. For the classification model we introduce another deterministic hierarchy with an equivalent parameterization as $\tilde{d}_{i,1}, ..., \tilde{d}_{i,M}$. We denote the hierarchy $\tilde{d}^{\text{c}}_{i,1}, ..., \tilde{d}^{\text{c}}_{i,M}$. The forward-pass is performed by:

$$\tilde{d}^{\text{C}}_{i,0} = \begin{cases} x & i = 1 \\ \tilde{d}^{\text{c}}_{i-1,M} & \text{otherwise} \end{cases} \tag{12}$$

$$\tilde{d}^{\text{C}}_{i,j} =< g^{\text{C}}_{\phi_{i,j}}(\tilde{d}^{\text{c}}_{i,j-1}); z^{\text{BU}}_i > \textbf{ for } j = 1, ..., M \tag{13}$$

$$y = g^{\text{c}}_{\phi_{i,M+1}}(\tilde{d}^{\text{c}}_{i,M}) , \tag{14}$$

where $g^{\text{c}}_{\phi_{i,M+1}}$ is a final densely connected neural network layer, of the same dimension as the number of categories, and a Softmax activation function. The inference model is thereby factorized by:

$$q_\phi(\mathbf{z}, y|x) = q_\phi(z_L|x, y, z^{\text{BU}}_{<L}) q_\phi(y|x, z^{\text{BU}}_{<L}) \prod_{i=1}^{L-1} q_\phi(z^{\text{BU}}_i|x, z^{\text{BU}}_{<i}) q_{\phi,\theta}(z^{\text{TD}}_i|x, y, z^{\text{BU}}_{<i}, z^{\text{BU}}_{>i}, z^{\text{TD}}_{>i}) . \tag{15}$$

Generative model. For each stochastic latent variable, $\mathbf{z}$, and the observed variable x in the generative model, as well as the TD path of the inference model, we add a conditional dependency on a categorical variable y:

$$p_\theta(x, y, \mathbf{z}) = p_\theta(x|\mathbf{z}, y) p_\theta(z_L) p_\theta(y) \prod_{i=1}^{L-1} p_\theta(z_i|z_{>i}, y) . \tag{16}$$

Evidence lower bound. In a semi-supervised learning problem, we have labeled data and unlabeled data which results in two formulations of the ELBO. The ELBO for labeled data points is given by:

$$\log p_\theta(x, y) \geq \mathbb{E}_{q_\phi(\mathbf{z}|x,y))}\left[\log \frac{p_\theta(x, y, \mathbf{z})}{q_{\phi,\theta}(\mathbf{z}|x, y)}\right] \equiv -\mathcal{F}(\theta, \phi) . \tag{17}$$

Since the classification model is not included in the above definition of the ELBO we add a classification loss term (a categorical cross-entropy), equivalent to the approach in [22]:

$$\bar{\mathcal{F}}(\theta, \phi) = \bar{\mathcal{F}}(\theta, \phi) - \alpha \cdot \mathbb{E}_{q(z<L|x)}[\log q_\phi(y|x, z^{\text{BU}}_{<L})] , \tag{18}$$

where α is a hyperparameter that we define as in [31]. For the unlabeled data points, we marginalize over the labels:

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(\mathbf{z},y|x)}\left[\log \frac{p_\theta(x, y, \mathbf{z})}{q_{\phi,\theta}(\mathbf{z}, y|x)}\right] \equiv -\mathcal{U}(\theta, \phi) . \tag{19}$$

The combined objective function over the labeled, (x_l, y_l), and unlabeled data points, (x_u), are thereby given by:

$$\mathcal{J}(\theta, \phi) = \sum_{x_l, y_l} \bar{\mathcal{F}}(\theta, \phi; x_l, y_l) + \sum_{x_u} \mathcal{U}(\theta, \phi; x_u) . \tag{20}$$

G Additional Results

Table 10: Test log-likelihood on dynamically binarized MNIST for different number of importance weighted samples. The finetuned models are trained for an additional number of epochs with no *free bits*, $\lambda = 0$.

	$-\log p(x)$
Results with autoregressive components	
DRAW+VGP [53]	< 79.88
IAFVAE [23]	≤ 79.10
VLAE [5]	≤ 78.53
Results without autoregressive components	
IWAE [4]	≤ 82.90
CONVVAE+HVI [47]	≤ 81.94
LVAE [50]	≤ 81.74
DISCRETE VAE [42]	≤ 80.04
BIVA, $\mathcal{L}_1$	≤ 80.60
BIVA, $\mathcal{L}_1 e3$	≤ 78.49
BIVA FINETUNED, $\mathcal{L}_1$	≤ 80.06
BIVA FINETUNED, $\mathcal{L}_{1e3}$	≤ 78.41

Table 11: Test log-likelihood on dynamically binarized OMNIGLOT for different number of importance weighted samples. The finetuned models are trained for an additional number of epochs with no *free bits*, $\lambda = 0$.

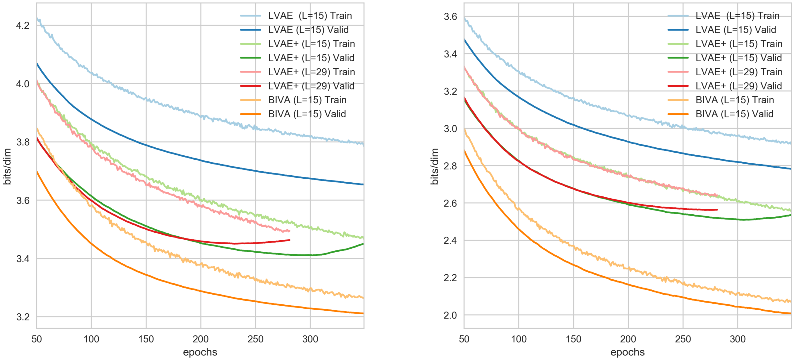
	$-\log p(x)$
Results with autoregressive components	
DRAW [13]	< 96.50
CONVDRAW [12]	< 91.00
VLAE [5]	≤ 89.83
Results without autoregressive components	
IWAE [4]	≤ 103.38
LVAE [50]	≤ 102.11
DVAE [42]	≤ 97.43
BIVA, $\mathcal{L}_1$	≤ 95.90
BIVA FINETUNED, $\mathcal{L}_1$	≤ 93.54
BIVA FINETUNED, $\mathcal{L}_{1e3}$	≤ 91.34

Table 12: Test log-likelihood on statically binarized Fashion MNIST for different number of importance weighted samples. The finetuned models are trained for an additional number of epochs with no *free bits*, $\lambda = 0$.

	$-\log p(x)$
BIVA, $\mathcal{L}_1$	≤ 94.05
BIVA FINETUNED, $\mathcal{L}_1$	≤ 93.54
BIVA FINETUNED, $\mathcal{L}_{1e3}$	≤ 87.98

Table 13: Test log-likelihood on ImageNet 32x32 for different number of importance weighted samples.

	BITS/DIM
With autoregressive components	
CONVDRAW [12]	< 4.10
PIXELRNN [57]	$= 3.63$
GATEDPIXELCNN [56]	$= 3.57$
Without autoregressive components	
REALNVP [9]	$= 4.28$
GLOW [21]	$= 4.09$
FLOW++ [16]	$= 3.86$
BIVA, $\mathcal{L}_1$	≤ 3.98
BIVA, $\mathcal{L}_{1e3}$	≤ 3.96



(a) $\mathcal{L}_1$ (bits/dim).

(b) $\log p_\theta(x|\mathbf{z})$ (bits/dim).

Figure 9: Convergence plot on CIFAR-10 training for the LVAE with $L = 15$, the LVAE+ with $L = 15$, the LVAE+ with $L = 29$, and BIVA with $L = 15$. (a) shows the convergence of the 1 importance weighted ELBO, $\mathcal{L}_1$, calculated in bits/dim. (b) shows the convergence of the *reconstruction loss*. The discrepancy between (a) and (b) is explained by the added cost from the stochastic latent variables, the Kullback-Leibler divergence $KL[p(\mathbf{z})||q(\mathbf{z}|x)]$.

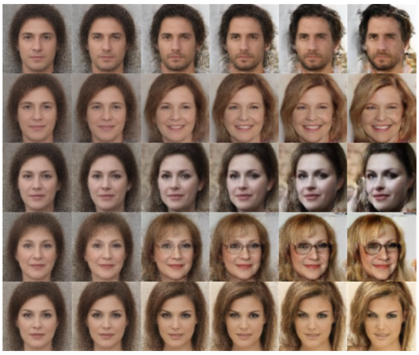
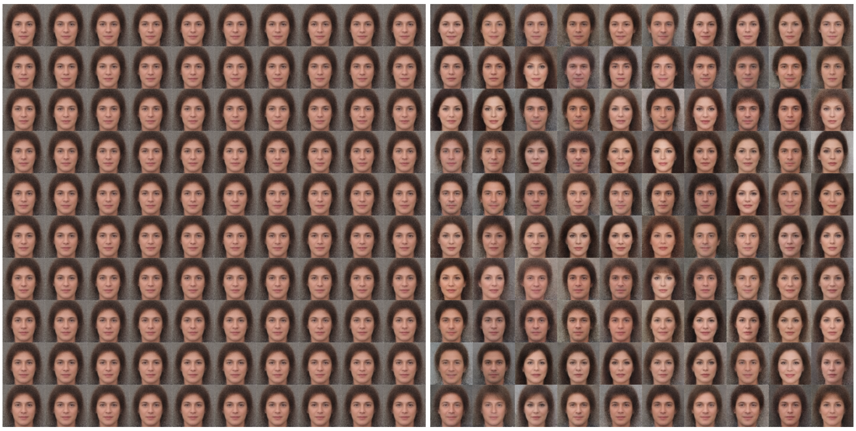
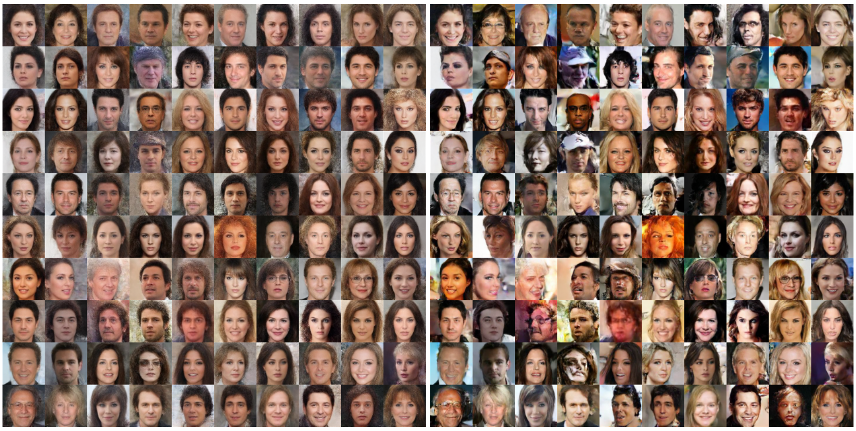


Figure 10: 64x64 CelebA samples generated from a BIVA with increasing levels of stochasticity in the model (going from close to the mode to the full distribution). In each column the latent variances are scaled with factors $0.1, 0.3, 0.5, 0.7, 0.9, 1.0$. Images in a row look similar because they use the same Gaussian random noise ϵ to generate the latent variables. BIVA has $L = 20$ stochastic latent layers connected by three layer ResNet blocks.



(a) $\sigma^2 = 0.01$ (b) $\sigma^2 = 0.1$



(c) $\sigma^2 = 0.5$ (d) $\sigma^2 = 1.0$

Figure 11: BIVA $\mathcal{N}(0, \sigma^2)$ generations with varying $\sigma^2 = 0.01, 0.1, 0.5, 1.0$ for (a), (b), (c) and (d) respectively. We follow the same generating procedure of Figure 10. BIVA has $L = 20$ stochastic latent variables and is trained on the CelebA dataset, preprocessed to 64x64 images following [27]. BIVA achieves a $\mathcal{L}_1 = 2.48$ bits/dim on the test set. Close to the mode of the latent distribution there is very little variance in generated natural images. When we *loosen* the samples towards the full distribution, $\sigma^2 = 1$, we can see how the generated images are adopting different styles and contexts.

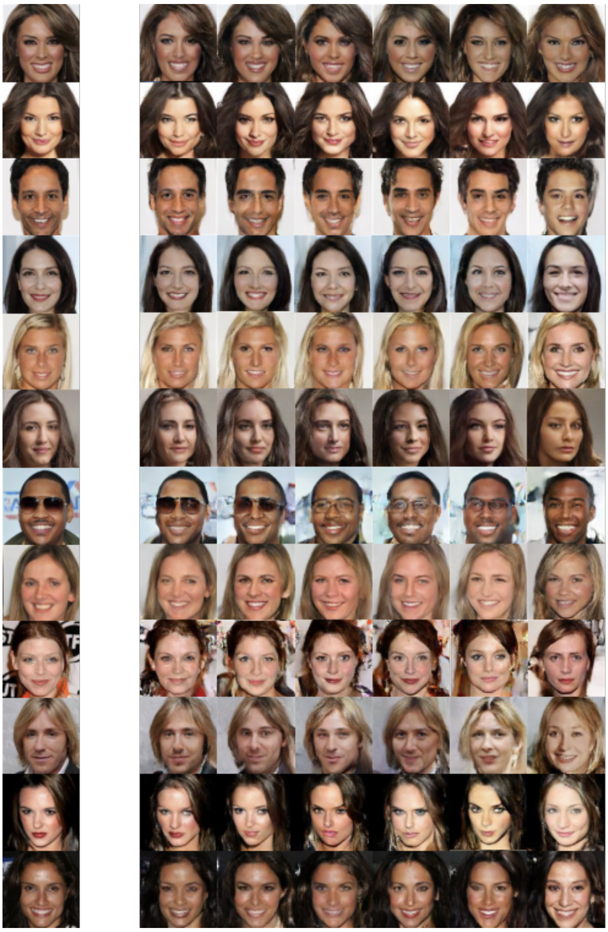


Figure 12: BIVA $L = 20$ generations (right) from fixed $z_{>i}$ given an input image (left), for different layers throughout the stochastic variable hierarchy (from left to right $i = 12, 14, 16, 17, 18, 19$). The model is trained on CelebA, preprocessed to 64x64 images following [27]. $z_{>i}$ are fixed by passing the original image through the encoder, after which $z_{\leq i}$ are sampled from the prior. When generating from a higher z_i (columns) it is shown how the model has more *freedom* to augment the input images. BIVA achieves a $\mathcal{L}_1 = 2.48$ bits/dim on the test set.

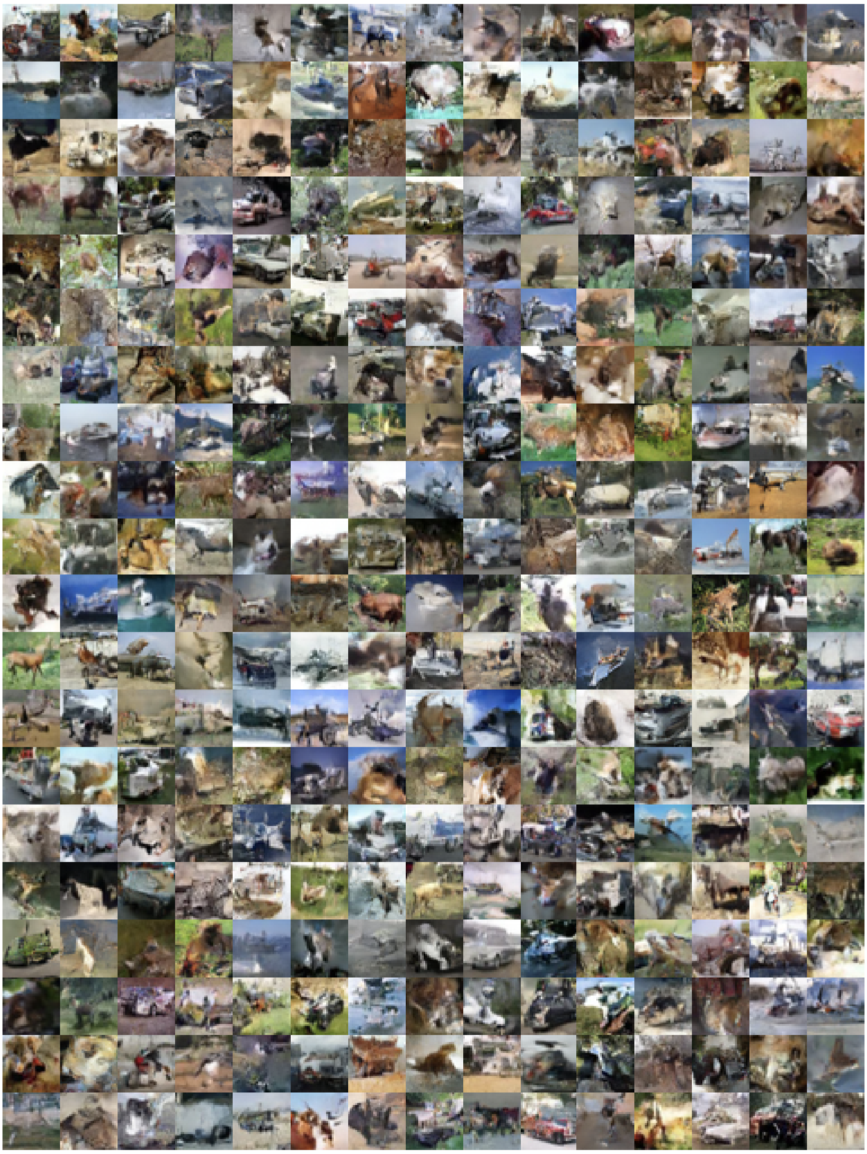


Figure 13: BIVA $\mathcal{N}(0, I)$ generations on a model trained on CIFAR-10. BIVA has $L = 15$ stochastic latent variables and achieves a 3.08 bits/dim on the test set. The images are still not as sharp and coherent as the PicelCNN++ [45] (3.08 vs. 2.92), however, it does achieve to find coherent structure resembling the categories of the CIFAR-10 dataset.

APPENDIX B

Optimal Variance Control of the Score-Function Gradient Estimator for Importance-Weighted Bounds

Optimal Variance Control of the Score Function Gradient Estimator for Importance Weighted Bounds

Valentin Liévin [1] **Andrea Dittadi**[1] **Anders Christensen**[1] **Ole Winther**[1, 2, 3]
[1] Section for Cognitive Systems, Technical University of Denmark
[2] Bioinformatics Centre, Department of Biology, University of Copenhagen
[3] Centre for Genomic Medicine, Rigshospitalet, Copenhagen University Hospital
`{valv,adit}@dtu.dk, anders.christensen321@gmail.com, olwi@dtu.dk`

Abstract

This paper introduces novel results for the score function gradient estimator of the importance weighted variational bound (IWAE). We prove that in the limit of large K (number of importance samples) one can choose the control variate such that the Signal-to-Noise ratio (SNR) of the estimator grows as $\sqrt{K}$. This is in contrast to the standard pathwise gradient estimator where the SNR decreases as $1/\sqrt{K}$. Based on our theoretical findings we develop a novel control variate that extends on VIMCO. Empirically, for the training of both continuous and discrete generative models, the proposed method yields superior variance reduction, resulting in an SNR for IWAE that increases with K without relying on the reparameterization trick. The novel estimator is competitive with state-of-the-art reparameterization-free gradient estimators such as Reweighted Wake-Sleep (RWS) and the thermodynamic variational objective (TVO) when training generative models.

1 Introduction

Gradient-based learning is now widespread in the field of machine learning, in which recent advances have mostly relied on the backpropagation algorithm, the workhorse of modern deep learning. In many instances, for example in the context of unsupervised learning, it is desirable to make models more expressive by introducing stochastic latent variables. Backpropagation thus has to be augmented with methodologies for marginalization over latent variables.

Variational inference using an inference model (amortized inference) has emerged as a key method for training and inference in latent variable models [1–7]. The pathwise gradient estimator, based on the reparameterization trick [2, 3], often gives low-variance estimates of the gradient for continuous distributions. However, since discrete distributions cannot be reparameterized, these methods are not applicable to inference in complex simulators with discrete variables, such as reinforcement learning or advanced generative processes [8–11]. While the score function (or Reinforce) estimator [12] is more generally applicable, it is well known to suffer from large variance. Consequently, most of the recent developments focus on reducing the variance using control variates [13–18] and using alternative variational objectives [9, 19–21].

Recently, variational objectives tighter than the traditional evidence lower bound (ELBO) have been proposed [21, 22]. In importance weighted autoencoders (IWAE) [22] the tighter bound comes with the price of a K-fold increase in the required number of samples from the inference network. Despite yielding a tighter bound, using more samples can be detrimental to the learning of the inference model [23]. In fact, the Signal-to-Noise ratio (the ratio of the expected gradient to its standard deviation) of the pathwise estimator has been shown to decrease at a rate $\mathcal{O}(K^{-1/2})$ [23]. Although this can be improved to $\mathcal{O}(K^{1/2})$ by exploiting properties of the gradient to cancel high-variance

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

terms [24], the variational distributions are still required to be reparameterizable. In this work we introduce OVIS (*Optimal Variance – Importance Sampling*), a novel score function-based estimator for importance weighted objectives with improved SNR.

The main contributions of this paper are: 1) A proof that, with an appropriate choice of control variate, the score function estimator for the IWAE objective can achieve a Signal-to-Noise Ratio $\mathrm{SNR} = \mathcal{O}(K^{1/2})$ as the number of importance samples $K \to \infty$. 2) A derivation of OVIS, a class of practical low-variance score function estimators following the principles of our theoretical analysis. 3) State-of-the-art results on a number of non-trivial benchmarks for both discrete and continuous stochastic variables, with comparison to a range of recently proposed score function methods.

2 Optimizing the Importance Weighted Bound

Importance weighted bound (IWAE) Amortized variational inference allows fitting a latent variable model $p_\theta(\mathbf{x}, \mathbf{z})$ to the data using an approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ [2]. By using multiple importance weighted samples, we can derive a lower bound to the log marginal likelihood that is uniformly tighter as the number of samples, K, increases [22]. The *importance weighted bound* (IWAE) for one data point $\mathbf{x}$ is:

$$\mathcal{L}_K(\mathbf{x}) := \mathbb{E}\left[\log \hat{Z}\right] \qquad \hat{Z} := \frac{1}{K}\sum_{k=1}^{K} w_k \qquad w_k := \frac{p_\theta(\mathbf{x}, \mathbf{z}_k)}{q_\phi(\mathbf{z}_k|\mathbf{x})}\ , \tag{1}$$

where $\mathbb{E}$ denotes an expectation over the K-copy variational posterior $q_\phi(\mathbf{z}_{1:K}|\mathbf{x}) := \prod_{k=1}^{K} q_\phi(\mathbf{z}_k|\mathbf{x})$. This bound coincides with the traditional evidence lower bound (ELBO) for $K = 1$. The log likelihood lower bound for the entire data set is $\mathcal{L}_K(\mathbf{x}_{1:n}) = \sum_{i=1}^{n} \mathcal{L}_K(\mathbf{x}_i)$. In the following we will derive results for one term $\mathcal{L}_K = \mathcal{L}_K(\mathbf{x})$.

Score function estimator Without making assumptions about the variational distribution, the gradient of the importance weighted bound (1) with respect to the parameters of the approximate posterior factorizes as (see Appendix A):

$$\nabla_\phi \mathcal{L}_K = \mathbb{E}\left[\sum\nolimits_k d_k \mathbf{h}_k\right] \qquad d_k := \log \hat{Z} - v_k \qquad v_k := \frac{w_k}{\sum_{l=1}^{K} w_l}\ , \tag{2}$$

where $\mathbf{h}_k := \nabla_\phi \log q_\phi(\mathbf{z}_k|\mathbf{x})$ is the score function. A Monte Carlo estimate of the expectation in (2) yields the *score function* (or *Reinforce*) *estimator*.

Control variates The vanilla score function estimator of (2) is often not useful in practice due to its large sample-to-sample variance. By introducing control variates that aim to cancel out zero expectation terms, this variance can be reduced while keeping the estimator unbiased.

Given posterior samples $\mathbf{z}_1, \ldots, \mathbf{z}_K \sim q_\phi(\mathbf{z}_{1:K}|\mathbf{x})$, let $\mathbf{z}_{-k}$ denote $[\mathbf{z}_1, \ldots, \mathbf{z}_{k-1}, \mathbf{z}_{k+1}, \ldots, \mathbf{z}_K]$, let $\mathbb{E}_k[\ldots]$ and $\mathbb{E}_{-k}[\ldots]$ be the expectations over the variational distributions of $\mathbf{z}_k$ and $\mathbf{z}_{-k}$, respectively, and let $\{c_k\}_{k=1}^{K}$ be scalar control variates, with each $c_k = c_k(\mathbf{z}_{-k})$ independent of $\mathbf{z}_k$. Using the independence of c_k and $\mathbf{h}_k$ for each k, and the fact that the score function has zero expectation, we have $\mathbb{E}[c_k \mathbf{h}_k] = \mathbb{E}_{-k}[c_k]\mathbb{E}_k[\mathbf{h}_k] = 0$. Thus, we can define an unbiased estimator of (2) as:

$$\mathbf{g} := \sum\nolimits_k (d_k - c_k)\,\mathbf{h}_k \tag{3}$$

$$\mathbb{E}[\mathbf{g}] = \mathbb{E}\left[\sum\nolimits_k (d_k - c_k)\,\mathbf{h}_k\right] = \mathbb{E}\left[\sum\nolimits_k d_k \mathbf{h}_k\right] = \nabla_\phi \mathcal{L}_K\ . \tag{4}$$

In the remainder of this paper, we will use the decomposition $d_k = f_k + f_{-k}$, where $f_k = f_k(\mathbf{z}_k, \mathbf{z}_{-k})$ and $f_{-k} = f_{-k}(\mathbf{z}_{-k})$ denote terms that depend and do not depend on $\mathbf{z}_k$, respectively. This will allow us to exploit the mutual independence of $\{\mathbf{z}_k\}_{k=1}^{K}$ to derive optimal control variates.

Signal-to-Noise Ratio (SNR) We will compare the different estimators on the basis of their Signal-to-noise ratio. Following [23], we define the SNR for each component of the gradient vector as

$$\mathrm{SNR}_i := \frac{|\mathbb{E}[g_i]|}{\sqrt{\mathrm{Var}[g_i]}}\ , \tag{5}$$

where g_i denotes the ith component of the gradient vector.

In Section 3 we derive the theoretical SNR for the optimal choice of control variates in the limit $K \to \infty$. In Section 4 we derive the optimal scalar control variates $\{c_k\}_{k=1}^K$ by optimizing the trace of the covariance of the gradient estimator $\mathbf{g}$, and in Section 6 we experimentally compare our approach with state-of-the-art gradient estimators in terms of SNR.

3 Asymptotic Analysis of the Signal-to-Noise Ratio

Assuming the importance weights have finite variance, i.e. $\mathrm{Var}[w_k] < \infty$, we can derive the asymptotic behavior of the SNR as $K \to \infty$ by expanding $\log \hat{Z}$ as a Taylor series around $Z := p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}, \mathbf{z}) d\mathbf{z}$ [23]. A direct application of the pathwise gradient estimator (reparameterization trick) to the importance weighted bound results in an SNR that scales as $\mathcal{O}(K^{-1/2})$ [23], which can be improved to $\mathcal{O}(K^{1/2})$ by exploiting properties of the gradient [24]. In the following we will show that, for a specific choice of control variate, the SNR of the score function estimator scales as $\mathcal{O}(K^{1/2})$. Thus, a score function estimator exists for which *increasing the number of importance samples benefits the gradient estimate of the parameters of the variational distribution.*

For the asymptotic analysis we rewrite the estimator as $\mathbf{g} = \sum_k \big(-\frac{\partial \log \hat{Z}}{\partial w_k} w_k + \log \hat{Z} - c_k \big) \mathbf{h}_k$ and apply a second-order Taylor expansion to $\log \hat{Z}$. The resulting expression $\mathbf{g} = \sum_k (f_k + f_{-k} - c_k)\mathbf{h}_k$ separates terms f_k that contribute to the expected gradient from terms f_{-k} that have zero expectation and thus only contribute to the variance (cf. Appendix B):

$$f_k \approx \frac{w_k^2}{2K^2Z^2} \tag{6}$$

$$f_{-k} \approx \log Z - \frac{3}{2} + \frac{2}{KZ} \sum\nolimits_{l \neq k} w_l - \frac{1}{2K^2Z^2} \Big(\sum\nolimits_{l \neq k} w_l \Big)^2 \, . \tag{7}$$

Since f_{-k} and c_k are independent of $\mathbf{h}_k$, the expected gradient is (cf. Appendix C.1):

$$\mathbb{E}[\mathbf{g}] = \sum\nolimits_k \mathbb{E}[f_k \mathbf{h}_k] \approx \frac{1}{2Z^2K} \mathbb{E}_1 \left[w_1^2 \mathbf{h}_1 \right] = \mathcal{O}(K^{-1}) \, , \tag{8}$$

where $\mathbb{E}_1$ denotes an expectation over the first latent distribution $q_\phi(\mathbf{z}_1|\mathbf{x})$. Since the choice of control variates $c_k = c_k(\mathbf{z}_{-k})$ is free, we can choose $c_k = f_{-k}$ to cancel out all zero expectation terms. The resulting covariance, derived in Appendix C.2, is:

$$\mathrm{Cov}[\mathbf{g}] = \mathrm{Cov} \left[\sum\nolimits_k f_k \mathbf{h}_k \right] \approx \frac{1}{4K^3Z^4} \mathrm{Cov}_1 \left[w_1^2 \mathbf{h}_1 \right] = \mathcal{O}(K^{-3}) \tag{9}$$

with Cov_1 indicating the covariance over $q_\phi(\mathbf{z}_1|\mathbf{x})$. Although as we discuss in Section 4 this is not the minimal variance choice of control variates, it is sufficient to achieve an SNR of $\mathcal{O}(K^{1/2})$.

4 Optimal Control Variate

The analysis above shows that in theory it is possible to attain a good SNR with the score function estimator. In this section we derive the optimal (in terms of variance of the resulting estimator) control variates $\{c_k\}_{k=1}^K$ by decomposing $\mathbf{g} = \sum_k (f_k + f_{-k} - c_k)\mathbf{h}_k$ as above, and minimizing the trace of the covariance matrix, i.e. $\mathbb{E}[||\mathbf{g}||^2] - ||\mathbb{E}[\mathbf{g}]||^2$. Since $\mathbb{E}[f_{-k}\mathbf{h}_k]$ and $\mathbb{E}[c_k\mathbf{h}_k]$ are both zero, $\mathbb{E}[\mathbf{g}] = \nabla_\phi \mathcal{L}_K$ does not depend on c_k. Thus, the minimization only involves the first term:

$$\begin{aligned} \frac{1}{2} \frac{\partial}{\partial c_k} \mathbb{E} \left[||\mathbf{g}||^2 \right] &= \mathbb{E} \left[\mathbf{h}_k^T \sum\nolimits_l (f_l + f_{-l} - c_l) \mathbf{h}_l \right] \\ &= \mathbb{E}_{-k} \left[\sum\nolimits_l \mathbb{E}_k \left[f_l \mathbf{h}_k^T \mathbf{h}_l \right] + (f_{-k} - c_k) \mathbb{E}_k \left[||\mathbf{h}_k||^2 \right] \right] \, . \end{aligned}$$

where $\mathbb{E}_k$ and $\mathbb{E}_{-k}$ indicate expectations over $q_\phi(\mathbf{z}_k|\mathbf{x})$ and $q_\phi(\mathbf{z}_{-k}|\mathbf{x})$, respectively. Setting the argument of $\mathbb{E}_{-k}$ to zero, we get the optimal control variates $c_k = c_k(\mathbf{z}_{-k})$ and gradient estimator $\mathbf{g}$:

$$c_k = f_{-k} + \sum\nolimits_l \frac{\mathbb{E}_k \left[f_l \mathbf{h}_k^T \mathbf{h}_l \right]}{\mathbb{E}_k \left[||\mathbf{h}_k||^2 \right]} \tag{10}$$

$$\mathbf{g} = \sum\nolimits_k \left(f_k - \sum\nolimits_l \frac{\mathbb{E}_k \left[f_l \mathbf{h}_k^T \mathbf{h}_l \right]}{\mathbb{E}_k \left[||\mathbf{h}_k||^2 \right]} \right) \mathbf{h}_k \, . \tag{11}$$

Applying (11) in practice requires marginalizing over one latent variable and decoupling terms that do not depend on $\mathbf{z}_k$ from those that do. In the remainder of this section we will 1) make a series of approximations to keep computation tractable, and 2) consider two limiting cases for the *effective sample size* (ESS) [25] in which we can decouple terms.

Simplifying approximations to Equation (11) First, we consider a term with $l \neq k$, define $\Delta f_l := f_l - \mathbb{E}_k[f_l]$, and subtract and add $\mathbb{E}_k[f_l]$ from inside the expectation:

$$\mathbb{E}_k\left[f_l \mathbf{h}_k^T\right]\mathbf{h}_l = \mathbb{E}_k\left[\Delta f_l \mathbf{h}_k^T\right]\mathbf{h}_l + \mathbb{E}_k[f_l]\mathbb{E}_k\left[\mathbf{h}_k^T\right]\mathbf{h}_l = \mathbb{E}_k\left[\Delta f_l \mathbf{h}_k^T\right]\mathbf{h}_l$$

where we used the fact that $\mathbb{E}_k\left[\mathbf{h}_k\right] = 0$. The $l \neq k$ terms thus only contribute to fluctuations relative to a mean value, and we assume they can be neglected.

Second, we assume that $|\phi|$, the number of parameters of q_ϕ, is large, and the terms of the sum $\|\mathbf{h}_k\|^2 = \sum_{i=1}^{|\phi|} h_{ki}^2$ are approximately independent with finite variances σ_i^2. By the Central Limit Theorem we approximate the distribution of $\Delta\|\mathbf{h}_k\|^2 := \|\mathbf{h}_k\|^2 - \mathbb{E}_k\left[\|\mathbf{h}_k\|^2\right]$ with a zero-mean Gaussian with standard deviation $\left(\sum_{i=1}^{|\phi|}\sigma_i^2\right)^{1/2}$. Seeing that $\mathbb{E}_k\left[\|\mathbf{h}_k\|^2\right]$ is $\mathcal{O}(|\phi|)$, we have

$$\frac{\mathbb{E}_k\left[f_k\|\mathbf{h}_k\|^2\right]}{\mathbb{E}_k\left[\|\mathbf{h}_k\|^2\right]} = \mathbb{E}_k\left[f_k\right] + \frac{\mathbb{E}_k\left[f_k\Delta\|\mathbf{h}_k\|^2\right]}{\mathbb{E}_k\left[\|\mathbf{h}_k\|^2\right]} = \mathbb{E}_k\left[f_k\right] + \mathcal{O}(|\phi|^{-1/2}) \,,$$

where we used that the argument in the numerator scales as $\left(\sum_{i=1}^{|\phi|}\sigma_i^2\right)^{1/2} = \mathcal{O}(|\phi|^{1/2})$.

Finally, the expectation can be approximated with a sample average. Writing $f_k = f_k(\mathbf{z}_k, \mathbf{z}_{-k})$ and drawing S new samples $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(S)} \sim q_\phi(\mathbf{z}|\mathbf{x})$:

$$\mathbb{E}_k\left[f_k\right] \approx \frac{1}{S}\sum_{s=1}^{S} f_k(\mathbf{z}^{(s)}, \mathbf{z}_{-k}) \,.$$

This will introduce additional fluctuations with scale $S^{-1/2}$.

Putting these three approximations together and using $d_k(\mathbf{z}_k, \mathbf{z}_{-k}) = f_k(\mathbf{z}_k, \mathbf{z}_{-k}) + f_{-k}(\mathbf{z}_{-k})$, we obtain the sample-based expression of the OVIS estimator, called $\mathrm{OVIS}_{\mathrm{MC}}$ in the following:

$$\mathrm{OVIS}_{\mathrm{MC}}: \quad \mathbf{g} \approx \sum_k \left(d_k(\mathbf{z}_k, \mathbf{z}_{-k}) - \frac{1}{S}\sum_{s=1}^{S} d_k(\mathbf{z}^{(s)}, \mathbf{z}_{-k}) \right)\mathbf{h}_k \,. \tag{12}$$

Naively, this will produce a large computational overhead because we now have in total KS terms. However, we can reduce this to $\mathcal{O}(K+S)$ because the bulk of the computation comes from evaluating the importance weights and because the S auxiliary samples can be reused for all K terms.

Effective sample size (ESS) The ESS [25] is a commonly used yardstick of the efficiency of an importance sampling estimate, defined as

$$\mathrm{ESS} := \frac{\left(\sum_k w_k\right)^2}{\sum_k w_k^2} = \frac{1}{\sum_k v_k^2} \in [1, K] \,. \tag{13}$$

A low ESS occurs when only a few weights dominate, which indicates that the proposal distribution q poorly matches p. In the opposite limit, the variance of importance weights is finite and the ESS will scale with K. Therefore the limit ESS $\gg 1$ corresponds to the asymptotic limit studied in Section 3.

Optimal control for ESS limits and unified interpolation In the following, we consider the two extreme limits ESS $\gg 1$ and ESS ≈ 1 to derive sample-free approximations to the optimal control. We can thus in these limits avoid the sample fluctuations and excess computation of $\mathrm{OVIS}_{\mathrm{MC}}$.

We first consider ESS $\gg 1$ and for each k we introduce the unnormalized leave-w_k-out approximation to $\hat{Z}$:

$$\widetilde{Z}_{[-k]} := \frac{1}{K}\sum_{l \neq k} w_l \quad \text{such that} \quad \hat{Z} - \widetilde{Z}_{[-k]} = \frac{w_k}{K} \,. \tag{14}$$

Assuming $\mathrm{Var}[w_k] < \infty$, this difference is $\mathcal{O}(K^{-1})$ as $K \to \infty$, thus we can expand $\log \hat{Z}$ around $\hat{Z} = \widetilde{Z}_{[-k]}$. In this limit, the optimal control variate simplifies to (cf. Appendix D.1):

$$\mathrm{ESS} \gg 1: \quad c_k \approx \log \frac{1}{K-1} \sum_{l \neq k} w_l + \log(1 - \frac{1}{K}) \,. \tag{15}$$

When ESS ≈ 1, one weight is much larger than the others and the assumption above is no longer valid. To analyze this frequently occurring scenario, assume that $k' = \operatorname{argmax}_l w_l$ and $w_{k'} \gg \sum_{l \neq k'} w_l$. In this limit $\log \hat{Z} \approx \log w_{k'}/K$ and $v_k \approx \delta_{k,k'}$ and thus $d_k = \log w_{k'}/K - \delta_{k,k'}$. In Appendix D.2 we show we can approximate Equation (10) with

$$\mathrm{ESS} \approx 1: \qquad c_k \approx \log \frac{1}{K-1} \sum_{l \neq k} w_l - v_k \,. \tag{16}$$

We introduce OVIS$_{\sim}$ to interpolate between the two limits (Appendix D.3):

$$c_k^{\gamma} := \log \frac{1}{K-1} \sum_{l \neq k} w_l - \gamma v_k + (1-\gamma) \log \left(1 - \frac{1}{K}\right) \qquad \gamma \in [0,1] \,. \tag{17}$$

In this paper we will only conduct experiments for the two limiting cases $\gamma = 0$, corresponding to Equation (15), and $\gamma = 1$ approximating Equation (16). Tuning the parameter γ in the range $[0, 1]$ will be left for future work. We discuss the implementation in the appendix K.

Higher ESS with looser lower bound Empirically we observe that training may be impaired by a low ESS and by *posterior collapse* [4, 26–29]. This motivates trading the tight IWAE objective for a gradient estimator with higher ESS. To that end, we use the importance weighted Rényi (IWR) bound:

$$\mathcal{L}_K^{\alpha}(\mathbf{x}) := \frac{1}{1-\alpha} \mathbb{E}\left[\log \hat{Z}(\alpha)\right] \quad \hat{Z}(\alpha) := \frac{1}{K} \sum\nolimits_k w_k^{1-\alpha} \tag{18}$$

which for $\alpha \in [0, 1]$ is a lower bound on the Rényi objective $\log \mathbb{E}_1 \left[w_1^{1-\alpha}\right] / (1-\alpha)$ [30]. The Rényi objective in itself coincides with $\log p(\mathbf{x})$ for $\alpha = 0$ and is monotonically non-increasing in α, i.e. is an evidence lower bound [30]. So we have a looser bound but higher $\mathrm{ESS}(\alpha) = 1/\sum_k v_k^2(\alpha) \geq \mathrm{ESS}(0)$ for $\alpha \in [0, 1]$ with $v_k(\alpha) = w_k^{1-\alpha} / \sum_l w_l^{1-\alpha}$. Furthermore, for $\alpha = 1$ the bound corresponds to the ELBO and the divergence $\mathcal{D}_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) || p_\theta(\mathbf{z}|\mathbf{x}))$ is guaranteed to be minimized. In Appendix E we derive the score function estimator and control variate expressions for $\mathcal{L}_K^{\alpha}$. The objective can either be used in a warm-up scheme by gradually decreasing $\alpha \to 0$ throughout iterations or can be run with a constant $0 < \alpha < 1$.

5 Related Work

The score function estimator with control variates can be used with all the commonly used variational families. By contrast, the reparameterization trick is only applicable under specific conditions. We now give a brief overview of the existing alternatives and refer the reader to [31] for a more extensive review. The importance of handling discrete distributions without relaxations is discussed in [9].

NVIL [13], DARN [17], and MuProp [18] demonstrate that score function estimators with carefully crafted control variates allow to train deep generative models. VIMCO [14] extends this to multi-sample objectives, and recycles the Monte Carlo samples $\mathbf{z}_{-k}$ to define a control variate $c_k = c_k(\mathbf{z}_{-k})$. Unlike OVIS, VIMCO only controls the variance of the term $\log \hat{Z}$ in $d_k = \log \hat{Z} - v_k$, leaving v_k uncontrolled, and causing the SNR to decrease with the number of particles K as we empirically observe in Section 6.1. We provide a detailed review of VIMCO in Appendix F.

The Reweighted Wake-Sleep (RWS) algorithm [20] is an extension of the original Wake-Sleep algorithm (ws) [19] that alternates between two distinct learning phases for optimizing importance weighted objectives. A detailed review of RWS and ws is available in Appendix F.

The Thermodynamic Variational Objective (TVO) [21] is a lower bound to $\log p_\theta(\mathbf{x})$ that stems from a Riemannian approximation of the Thermodynamic Variational Identity (TVI), and unifies the objectives of Variational Inference and Wake-Sleep. Evaluating the gradient involves differentiating through an expectation over a distribution with an intractable normalizing constant. To accommodate this, the authors propose an estimator that generalizes the score function estimator based on a tractable covariance term. We review the TVO in more detail in Appendix F.

Given a deterministic *sampling path* $g(\boldsymbol{\epsilon}; \theta)$ such that $\mathbf{z} \sim p_\theta(\mathbf{z})$ and $\mathbf{z} = g(\boldsymbol{\epsilon}; \theta), \boldsymbol{\epsilon} \sim p(\boldsymbol{\epsilon})$ are equivalent, one can derive a *pathwise gradient estimator* of the form $\nabla_\theta \mathbb{E}_{p_\theta(\mathbf{z})}[f_\theta(\mathbf{z})] = \mathbb{E}_{p(\boldsymbol{\epsilon})}[\nabla_\theta f_\theta(g(\boldsymbol{\epsilon}; \theta))]$. This estimator – introduced in machine learning as the *reparameterization trick* or *stochastic backpropagation* [2, 3] – exhibits low variance thanks to the structural information provided by the sampling path. Notably, a zero expectation term can be removed from the estimator [32]. Extending on this, [24] derives an alternative gradient estimator for IWAE that exhibits SNR $\sim K^{1/2}$, as opposed to SNR $\sim K^{-1/2}$ for the *standard* IWAE objective [23].

Continuous relaxations of discrete distributions yield a biased low-variance gradient estimate thanks to the reparameterization trick [16, 33]. Discrete samples can be obtained using the Straight-Through estimator [5, 34]. The resulting gradient estimate remains biased, but can be used as a control variate for the score function objective, resulting in an unbiased low-variance estimate of the gradient [15, 35].

6 Experimental Results

We conduct a number of experiments[1] on benchmarks that have previously been used to test score function based estimators. All models are trained via stochastic gradient ascent using the Adam optimizer [36] with default parameters. We use regular gradients on the training objective for the generative model parameters θ. The SNR for θ scales as $\mathcal{O}(K^{1/2})$ [23].

6.1 Asymptotic Variance

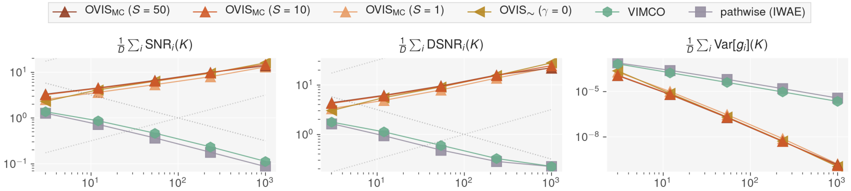


Figure 1: Gaussian model. Parameter-wise average of the asymptotic SNR, DSNR and variance of the gradients of the parameter b for different number of particles $K \in [3, 1000]$ using 10^4 MC samples. The dotted lines stand for $y = 10^{\pm 1} K^{\pm 0.5}$.

Following [23], we empirically corroborate the asymptotic properties of the OVIS gradient estimator by means of the following simple model:

$$\mathbf{z} \sim \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}, \mathbf{I}), \quad \mathbf{x}|\mathbf{z} \sim \mathcal{N}(\mathbf{x}; \mathbf{z}, \mathbf{I}), \quad q_\phi(\mathbf{z}|\mathbf{x}) = \mathcal{N}\left(\mathbf{z}; \mathbf{A}\mathbf{x} + \mathbf{b}, \tfrac{2}{3}\mathbf{I}\right).$$

where $\mathbf{x}$ and $\mathbf{z}$ are real vectors of size $D = 20$. We sample $N = 1024$ points $\left\{\mathbf{x}^{(n)}\right\}_{n=1}^{N}$ from the *true* model where $\boldsymbol{\mu}^\star \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The optimal parameters are $\mathbf{A}^\star = \mathbf{I}/2$, $\mathbf{b}^\star = \boldsymbol{\mu}^\star/2$, and $\boldsymbol{\mu}^\star = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}^{(n)}$. The model parameters are obtained by adding Gaussian noise of scale $\epsilon = 10^{-3}$. We measure the variance and the SNR of the gradients with 10^4 MC samples. We also measured the *directional* SNR (DSNR [23]) to probe if our results hold in the multidimensional case.

In Figure 1 we report the gradient statistics for $\mathbf{b}$. We observe that using more samples in the standard IWAE leads to a decrease in SNR as $\mathcal{O}(K^{-1/2})$ for both VIMCO and the pathwise-IWAE [23]. The tighter variance control provided by OVIS leads the variance to decrease almost at a rate $\mathcal{O}(K^{-3})$, resulting in a measured SNR not far from $\mathcal{O}(K^{1/2})$ both for OVIS$_{\mathrm{MC}}$ and OVIS$_{\sim}$. This shows that, despite the approximations, the proposed gradient estimators OVIS$_{\mathrm{MC}}$ and OVIS$_{\sim}$ are capable of achieving the theoretical SNR of $\mathcal{O}(K^{1/2})$ derived in the asymptotic analysis in Section 3.

---

[1] The full experimental framework is available at github.com/vlievin/ovis

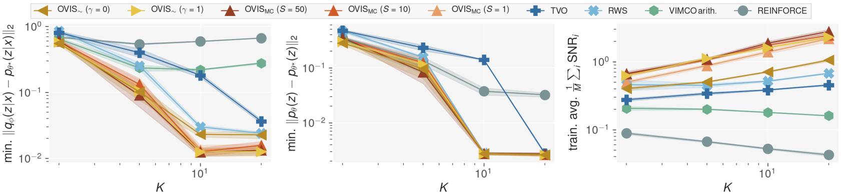


Figure 2: Training of the Gaussian mixture model. Minimum test-diagnostics recorded during training and training average of the SNR of the gradients of ϕ with $M = \text{card}(\phi)$. In contrast to VIMCO, OVIS$_\sim$ and OVIS$_{\text{MC}}$ all benefit from the increase of the particles budget, OVIS$_{\text{MC}}$ yields the most accurate posterior among the compared methods.

In Appendix G, we learn the parameters of the Gaussian model using OVIS, RWS, VIMCO and the TVO. We find that optimal variance reduction translates into a more accurate estimation of the optimal parameters of the inference network when compared to RWS, VIMCO and the TVO.

6.2 Gaussian Mixture Model

We evaluate OVIS on a Gaussian Mixture Model and show that, unlike VIMCO [9], our method yields better inference networks as the number of particles K increases. Following [9], we define:

$$p_\theta(z) = \text{Cat}(z|\,\text{softmax}(\theta)) \quad p(x|z) = \mathcal{N}\left(x|\mu_z, \sigma_z^2\right) \quad q_\phi(z|x) = \text{Cat}\left(z|\,\text{softmax}\left(\eta_\phi(x)\right)\right)$$

where $z \in \{0, \ldots, C-1\}$, $\mu_z = 10z$, $\sigma_z = 5$, and $C = 20$ is the number of clusters. The inference network η_ϕ is parameterized by a multilayer perceptron with architecture 1–16–C and $\tanh$ activations. The true generative model is set to $p_{\theta^\star}(z = c) = (c+5)/\sum_{i=1}^{C}(i+5)$.

All models are trained for 100k steps with 5 random seeds. We compare OVIS with VIMCO, RWS with wake-ϕ update, Reinforce, and the TVO. For the latter we chose to use 5 partitions and $\beta_1 = 10^{-2}$, after a hyperparameter search over $\beta_1 \in \{10^{-1}, 10^{-1.5}, 10^{-2}, 10^{-2.5}, 10^{-3}\}$ and $\{2, 5\}$ partitions.

Each model is evaluated on a held-out test set of size $M = 100$. We measure the accuracy of the learned posterior $q_\phi(z|x)$ by its average L_2 distance from the true posterior, i.e. $\frac{1}{M}\sum_{m=1}^{M}\left\|q_\phi\left(z|x^{(m)}\right) - p_{\theta^\star}\left(z|x^{(m)}\right)\right\|_2$. As a sanity check, we assess the quality of the generative model using $\|\text{softmax}(\theta) - \text{softmax}\left(\theta^\star\right)\|_2$. The SNR of the gradients for the parameters ϕ is evaluated on one mini-batch of data using 500 MC samples.

We report our main results in Figure 2, and training curves in Appendix H. In contrast to VIMCO, the accuracy of the posteriors learned using OVIS$_{\text{MC}}$ and OVIS$_\sim$ all improve monotonically with K and outperform the baseline estimators, independently of the choice of the number of auxiliary particles S. All OVIS methods outperform the state-of-the-art estimators RWS and the TVO, as measured by the L_2 distance between the approximate and the true posterior.

6.3 Deep Generative Models

We utilize the OVIS estimators to learn the parameters of both discrete and continuous deep generative models using stochastic gradient ascent. The base learning rate is fixed to $3 \cdot 10^{-4}$, we use mini-batches of size 24 and train all models for $4 \cdot 10^6$ steps. We use the statically binarized MNIST dataset [37] with the original training/validation/test splits of size 50k/10k/10k. We follow the experimental protocol as detailed in [21], including the β partition for the TVO and the exact architecture of the models. We use a three-layer Sigmoid Belief Network [38] as an archetype of discrete generative model [13, 14, 21] and a Gaussian Variational Autoencoder [2] with 200 latent variables. All models are trained with three initial random seeds and for $K \in \{5, 10, 50\}$ particles.

We assess the performance based on the marginal log-likelihood estimate $\log \hat{p}_\theta(\mathbf{x}) = \mathcal{L}_{5000}(\mathbf{x})$, that we evaluate on 10k *training* data points, such as to disentangle the training dynamics from the regularisation effect that is specific to each method. We measure the quality of the inference network solution using the divergence $\mathcal{D}_{\text{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})\right) \approx \log \hat{p}_\theta(\mathbf{x}) - \mathcal{L}_1(\mathbf{x})$. The full training curves

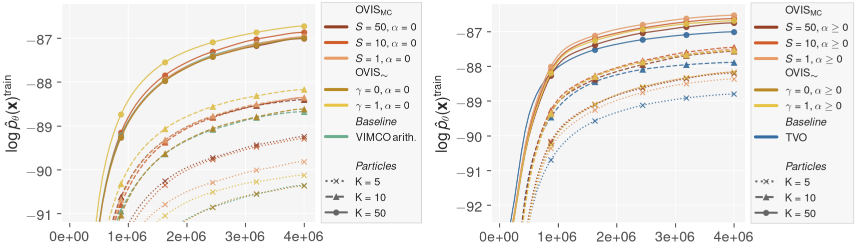


Figure 3: Training a Sigmoid Belief Network on Binarized MNIST. (Left) Optimizing for the importance weighted bound $\mathcal{L}_K$ using OVIS. (Right) Optimizing for the Rényi importance lower bound $\mathcal{L}_K^\alpha$ using OVIS with α annealing $0.99 \rightarrow 0$. The curves are averaged over three seeds and smoothed for clarity.

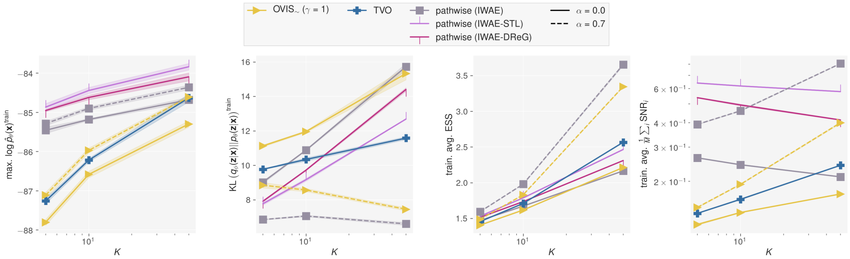


Figure 4: Training a one layer Gaussian VAE. Maximum recorded training $\log \hat{p}_\theta(\mathbf{x})$, final estimate of the bound $\mathcal{D}_{\text{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})\right)$ and training average of the ESS and of the SNR. OVIS yields similar likelihood performances as the TVO but benefits from a tighter bound thanks to optimizing for the IWR bound.

– including the test log likelihood and divergences – are available in Appendix J. We will show that OVIS improves over VIMCO, on which it extends, and we show that combining OVIS$_\sim$ with the Variational Rényi bound (IWR) as described in Section 4 outperforms the TVO.

6.3.1 Sigmoid Belief Network (SBN)

A. Comparison with VIMCO We learn the parameters of the SBN using the OVIS estimators for the IWAE bound and use VIMCO as a baseline. We report $\log \hat{p}_\theta(\mathbf{x})$ in the left plot of Figure 3. All OVIS methods outperform VIMCO, ergo supporting the advantage of optimal variance reduction. When using a small number of particles $K = 5$, learning can be greatly improved by using an accurate MC estimate of the optimal control variate, as suggested by OVIS$_{\text{MC}}(S = 50)$ which allows gaining $+1.0$ nats over VIMCO. While OVIS$(\gamma = 0)$, designed for large ESS barely improved over VIMCO, the biased OVIS$_\sim(\gamma = 1)$ for low ESS performed significantly better than other methods for $K \geq 10$, which coincides with the ESS measured in the range $[1.0, 3.5]$ for all methods. We attribute the relative decrease of performances observed for OVIS$_{\text{MC}}$ for $K = 50$ to *posterior collapse*.

B. Training using IWR bounds In Figure 3 (right) we train the SBN using OVIS and the TVO. OVIS is coupled with the objective $\mathcal{L}_K^\alpha$ for which we anneal the parameter α from 0.99 ($\mathcal{L}_K^{0.99} \approx \mathcal{L}_1$) to 0 ($\mathcal{L}_K^0 = \mathcal{L}_K$) during $1e6$ steps using geometric interpolation. For all K values, OVIS outperform the TVO and OVIS$_\sim(\gamma = 1)$ performs comparably with OVIS$_{\text{MC}}$.

6.3.2 Gaussian Variational Autoencoder (VAE)

In Figure 4 we train the Gaussian VAE using the standard pathwise IWAE, Sticking the Landing (STL) [32], DReG [24], the TVO and $\mathrm{OVIS}_{\sim}(\gamma = 1)$.

OVIS is applied to the IWR bound with $\alpha = 0.7$. As measured by the training likelihood, $\mathrm{OVIS}_{\sim}(\gamma = 1)$ coupled with the IWR bound performs on par with the TVO, which bridges the gap to the standard pathwise IWAE for $K = 50$, although different objectives are at play. The advanced pathwise estimators (STL and DReG) outperform all other methods. Measuring the quality of the learned proposals $q_\phi(\mathbf{z}|\mathbf{x})$ using the KL divergence allows disentangling the TVO and $\mathrm{OVIS}_{\sim}$ methods, as $\mathrm{OVIS}(\gamma = 1)$ applied to the IWR bound outputs higher-quality approximate posteriors for all considered number of particles.

6.4 A final Note on $\mathrm{OVIS}_{\sim}(\gamma = 1)$

$\mathrm{OVIS}_{\sim}(\gamma = 1)$ generates training dynamics that are superior to the baseline TVO and to $\mathrm{OVIS}_{\mathrm{MC}}$ given a comparable particle budget (appendix I). We interpret this result as a consequence of the ESS-specific design, which also appeared to be robust to the choice of α in the IWR objective. This also corroborates the results of [32], that suppressing the term $-\sum_k v_k \mathbf{h}_k$ from the gradient estimate improves learning. We therefore recommend the practitioner to first experiment with $\mathrm{OVIS}_{\sim}(\gamma = 1)$ since it delivers competitive results at a reasonable computational cost.

7 Conclusion

We proposed OVIS, a gradient estimator that is generally applicable to deep models with stochastic variables, and is empirically shown to have optimal variance control. This property is achieved by identifying and canceling terms in the estimator that solely contribute to the variance. We expect that in practice it will often be a good trade-off to use a looser bound with a higher effective sample size, e.g. by utilizing the OVIS estimator with the importance weighted Rényi bound, allowing control of this trade-off via an additional scalar smoothing parameter. This sentiment is supported by our method demonstrating better performance than the current state-of-the-art.

8 Financial Disclosure

The PhD program supporting Valentin Liévin is partially funded by Google. This research was supported by the NVIDIA Corporation with the donation of GPUs.

As with other theoretical advances such as those presented in this paper, consequences are not immediate and depend on the applications in which the research is utilized. It is our hope that this research will ultimately be of practical use with a tangible positive impact.

References

[1] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[2] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[3] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

[4] Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*, 2015.

[5] Aaron van den Oord, Oriol Vinyals, et al. Neural discrete representation learning. In *Advances in Neural Information Processing Systems*, pages 6306–6315, 2017.

[6] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *Iclr*, 2(5):6, 2017.

[7] Lars Maaløe, Marco Fraccaro, Valentin Liévin, and Ole Winther. BIVA: A very deep hierarchy of latent variables for generative modeling. In *Advances in neural information processing systems*, pages 6548–6558, 2019.

[8] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pages 1057–1063, 2000.

[9] Tuan Anh Le, Adam R Kosiorek, N Siddharth, Yee Whye Teh, and Frank Wood. Revisiting reweighted wake-sleep. *arXiv preprint arXiv:1805.10469*, 2018.

[10] SM Ali Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, David Szepesvari, Geoffrey E Hinton, et al. Attend, infer, repeat: Fast scene understanding with generative models. In *Advances in Neural Information Processing Systems*, pages 3225–3233, 2016.

[11] Yishu Miao and Phil Blunsom. Language as a latent variable: Discrete generative models for sentence compression. *arXiv preprint arXiv:1609.07317*, 2016.

[12] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

[13] Andriy Mnih and Karol Gregor. Neural variational inference and learning in belief networks. *arXiv preprint arXiv:1402.0030*, 2014.

[14] Andriy Mnih and Danilo J Rezende. Variational inference for monte carlo objectives. *arXiv preprint arXiv:1602.06725*, 2016.

[15] George Tucker, Andriy Mnih, Chris J Maddison, John Lawson, and Jascha Sohl-Dickstein. Rebar: Low-variance, unbiased gradient estimates for discrete latent variable models. In *Advances in Neural Information Processing Systems*, pages 2627–2636, 2017.

[16] Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

[17] Karol Gregor, Ivo Danihelka, Andriy Mnih, Charles Blundell, and Daan Wierstra. Deep autoregressive networks. *arXiv preprint arXiv:1310.8499*, 2013.

[18] Shixiang Gu, Sergey Levine, Ilya Sutskever, and Andriy Mnih. Muprop: Unbiased backpropagation for stochastic neural networks. *arXiv preprint arXiv:1511.05176*, 2015.

[19] Geoffrey E Hinton, Peter Dayan, Brendan J Frey, and Radford M Neal. The" wake-sleep" algorithm for unsupervised neural networks. *Science*, 268(5214):1158–1161, 1995.

[20] Jörg Bornschein and Yoshua Bengio. Reweighted wake-sleep. *arXiv preprint arXiv:1406.2751*, 2014.

[21] Vaden Masrani, Tuan Anh Le, and Frank Wood. The thermodynamic variational objective. In *Advances in Neural Information Processing Systems*, pages 11521–11530, 2019.

[22] Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*, 2015.

[23] Tom Rainforth, Adam R Kosiorek, Tuan Anh Le, Chris J Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh. Tighter variational bounds are not necessarily better. *arXiv preprint arXiv:1802.04537*, 2018.

[24] George Tucker, Dieterich Lawson, Shixiang Gu, and Chris J Maddison. Doubly reparameterized gradient estimators for monte carlo objectives. *arXiv preprint arXiv:1810.04152*, 2018.

[25] Augustine Kong. A note on importance sampling using standardized weights. *University of Chicago, Dept. of Statistics, Tech. Rep*, 348, 1992.

[26] Xi Chen, Diederik P Kingma, Tim Salimans, Yan Duan, Prafulla Dhariwal, John Schulman, Ilya Sutskever, and Pieter Abbeel. Variational lossy autoencoder. *arXiv preprint arXiv:1611.02731*, 2016.

[27] Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. In *Advances in neural information processing systems*, pages 3738–3746, 2016.

[28] Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. In *Advances in neural information processing systems*, pages 4743–4751, 2016.

[29] Adji B Dieng, Yoon Kim, Alexander M Rush, and David M Blei. Avoiding latent variable collapse with generative skip models. *arXiv preprint arXiv:1807.04863*, 2018.

[30] Yingzhen Li and Richard E Turner. Rényi divergence variational inference. In *Advances in Neural Information Processing Systems*, pages 1073–1081, 2016.

[31] Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. Monte carlo gradient estimation in machine learning. *arXiv preprint arXiv:1906.10652*, 2019.

[32] Geoffrey Roeder, Yuhuai Wu, and David K Duvenaud. Sticking the landing: Simple, lower-variance gradient estimators for variational inference. In *Advances in Neural Information Processing Systems*, pages 6925–6934, 2017.

[33] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[34] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

[35] Will Grathwohl, Dami Choi, Yuhuai Wu, Geoffrey Roeder, and David Duvenaud. Backpropagation through the void: Optimizing control variates for black-box gradient estimation. *arXiv preprint arXiv:1711.00123*, 2017.

[36] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[37] Ruslan Salakhutdinov and Iain Murray. On the quantitative analysis of deep belief networks. In *Proceedings of the 25th international conference on Machine learning*, pages 872–879, 2008.

[38] Radford M Neal. Connectionist learning of belief networks. *Artificial intelligence*, 56(1): 71–113, 1992.

A Derivation of the Score Function Estimator

Given K samples, the objective being maximized is

$$\mathcal{L}_K(\mathbf{x}) := \mathbb{E}\left[\log \hat{Z}\right] \qquad \hat{Z} := \frac{1}{K}\sum_{k=1}^{K} w_k \qquad w_k := \frac{p_\theta(\mathbf{x}, \mathbf{z}_k)}{q_\phi(\mathbf{z}_k|\mathbf{x})} \, . \tag{19}$$

The gradients of the multi-sample objective $\mathcal{L}_K$ with respect to the parameter ϕ can be expressed as a sum of two terms, one arising from the expectation over the variational posterior $q_\phi(\mathbf{z}_{1:K}|\mathbf{x}) := \prod_{k=1}^{K} q_\phi(\mathbf{z}_k|\mathbf{x})$ and one from $\log \hat{Z}$:

$$\nabla_\phi \mathcal{L}_K = \underbrace{\mathbb{E}\left[\log \hat{Z} \frac{\nabla_\phi q_\phi(\mathbf{z}_{1:K}|\mathbf{x})}{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})}\right]}_{\textbf{(a)}} + \underbrace{\mathbb{E}\left[\nabla_\phi \log \hat{Z}\right]}_{\textbf{(b)}} \, .$$

The term **(a)** yields the traditional score function estimator

$$\begin{aligned}\textbf{(a)} &= \mathbb{E}\left[\log \hat{Z}\, \nabla_\phi \log q_\phi(\mathbf{z}_{1:K}|\mathbf{x})\right] \\ &= \mathbb{E}\left[\log \hat{Z} \sum_{k=1}^{K} \nabla_\phi \log q_\phi(\mathbf{z}_k|\mathbf{x})\right] \, .\end{aligned} \tag{20}$$

The term **(b)** is

$$\begin{aligned}\textbf{(b)} &= \mathbb{E}\left[\nabla_\phi \log \frac{1}{K}\sum_{k=1}^{K} w_k\right] \\ &= \mathbb{E}\left[\frac{1}{\frac{1}{K}\sum_{k=1}^{K} w_k} \nabla_\phi \frac{1}{K}\sum_{k=1}^{K} w_k\right] \\ &= \mathbb{E}\left[\frac{1}{\sum_{l=1}^{K} w_l} \sum_{k=1}^{K} \nabla_\phi w_k\right] \\ &= \mathbb{E}\left[\frac{1}{\sum_{l=1}^{K} w_l} \sum_{k=1}^{K} w_k \nabla_\phi \log w_k\right] \\ &= \mathbb{E}\left[\sum_{k=1}^{K} v_k \nabla_\phi \log w_k\right] , \quad v_k = \frac{w_k}{\sum_{l=1}^{K} w_l} \\ &= -\mathbb{E}\left[\sum_{k=1}^{K} v_k \nabla_\phi \log q_\phi(\mathbf{z}_k|\mathbf{x})\right] \, .\end{aligned} \tag{21}$$

The derivation yields a factorized expression of the gradients

$$\nabla_\phi \mathcal{L}_K = \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})}\left[\sum_{k=1}^{K}\left(\log \hat{Z} - v_k\right)\mathbf{h}_k\right] \qquad \text{with} \qquad \mathbf{h}_k := \nabla_\phi \log q_\phi(\mathbf{z}_k|\mathbf{x}) \, . \tag{22}$$

B Asymptotic Analysis

We present here a short derivation and direct the reader to [23] for the fine prints of the proof. The main requirement is that w_k is bounded, so that $\hat{Z} - Z$ (with $Z = p(\mathbf{x})$) will converge to 0 almost surely as $K \to \infty$. We can also state this through the central limit theorem by noting that $\hat{Z} - Z = \frac{1}{K}\sum_k (w_k - Z)$ is the sum of K independent terms so if $\mathrm{Var}_1[w_1]$ is finite then $\hat{Z} - Z$ will converge to a Gaussian distribution with mean $\mathbb{E}[\hat{Z} - Z] = 0$ and variance $\mathrm{Var}[\hat{Z} - Z] = \frac{1}{K}\mathrm{Var}_1[w_1]$. The K^{-1} factor on the variance follows from independence. This means that in a Taylor expansion in $\hat{Z} - Z$ higher order terms will be suppressed.

Rewriting $\mathbf{g}$ in terms of $\log \hat{Z}$:

$$\mathbf{g} = \sum_k (d_k - c_k)\,\mathbf{h}_k = \sum_k \left(\log \hat{Z} - w_k \frac{\partial}{\partial w_k} \log \hat{Z} - c_k\right) \mathbf{h}_k \tag{23}$$

and using the second-order Taylor expansion of $\log \hat{Z}$ about Z:

$$\log \hat{Z} \approx \log Z + \frac{\hat{Z} - Z}{Z} - \frac{(\hat{Z} - Z)^2}{2Z^2} \tag{24}$$

we have

$$\log \hat{Z} \approx \log Z - \frac{3}{2} + \frac{2}{KZ} \sum_l w_l - \frac{1}{2K^2 Z^2} \left(\sum_l w_l\right)^2 \tag{25}$$

$$\frac{\partial}{\partial w_k} \log \hat{Z} \approx \frac{2}{KZ} - \frac{1}{K^2 Z^2} \sum_l w_l \,. \tag{26}$$

The term d_k can thus be approximated as follows:

$$\begin{aligned} d_k &= \log \hat{Z} - w_k \frac{\partial}{\partial w_k} \log \hat{Z} \\ &\approx \log Z - \frac{3}{2} + \frac{2}{KZ} \sum_{l \neq k} w_l - \frac{1}{2K^2 Z^2} \left(\sum_l w_l\right)^2 + \frac{1}{K^2 Z^2} w_k \sum_l w_l \\ &= \log Z - \frac{3}{2} + \frac{2}{KZ} \sum_{l \neq k} w_l - \frac{1}{2K^2 Z^2} \left(\sum_{l \neq k} w_l\right)^2 + \frac{1}{2K^2 Z^2} w_k^2 \end{aligned} \tag{27}$$

where we used

$$\left(\sum_l w_l\right)^2 = \left(\sum_{l \neq k} w_l\right)^2 + w_k^2 + 2 w_k \sum_{l \neq k} w_l \,.$$

By separately collecting the terms that depend and do not depend on $\mathbf{z}_k$ into $f_k = f_k(\mathbf{z}_k, \mathbf{z}_{-k})$ and $f_{-k} = f_{-k}(\mathbf{z}_{-k})$, respectively, we can rewrite the estimator $\mathbf{g}$ as:

$$\mathbf{g} = \sum_k (f_k + f_{-k} - c_k) \mathbf{h}_k \tag{28}$$

and from (27) we have

$$f_k \approx \frac{w_k^2}{2K^2 Z^2} \tag{29}$$

$$f_{-k} \approx \log Z - \frac{3}{2} + \frac{2}{KZ} \sum_{l \neq k} w_l - \frac{1}{2K^2 Z^2} \left(\sum_{l \neq k} w_l\right)^2 . \tag{30}$$

C Asymptotic Expectation and Variance

We derive here the asymptotic expectation and variance of the gradient estimator $\mathbf{g}$ in the limit $K \to \infty$.

C.1 Expectation

If both f_{-k} and c_k are independent of $\mathbf{z}_k$, we can write:

$$\mathbb{E}[\mathbf{g}] = \mathbb{E}\left[\sum_k (f_k + f_{-k} - c_k) \mathbf{h}_k\right] = \sum_k \mathbb{E}\left[f_k \mathbf{h}_k\right] \tag{31}$$

where we used that $\mathbb{E}\left[f_{-k}\mathbf{h}_k\right]$ and $\mathbb{E}\left[c_k\mathbf{h}_k\right]$ are zero. In the limit $K \to \infty$, each term of the sum can be expanded with the approximation (29) and simplified:

$$\mathbb{E}\left[f_k\mathbf{h}_k\right] \approx \mathbb{E}\left[\frac{w_k^2}{2K^2Z^2}\mathbf{h}_k\right] = \frac{1}{2K^2Z^2}\mathbb{E}_1\left[w_1^2\mathbf{h}_1\right] \tag{32}$$

where $\mathbb{E}_1$ denotes an expectation over the posterior $q_\phi(\mathbf{z}_1|\mathbf{x})$. The last step follows from the fact that the latent variables $\{\mathbf{z}_k\}_{k=1}^K$ are i.i.d. and the argument of the expectation only depends on one of them. In conclusion, the expectation is:

$$\mathbb{E}[\mathbf{g}] = \sum_k \mathbb{E}\left[f_k\mathbf{h}_k\right] \approx \frac{1}{2KZ^2}\mathbb{E}_1\left[w_1^2\mathbf{h}_1\right] = \mathcal{O}(K^{-1}) \tag{33}$$

irrespective of f_{-k} and c_k.

C.2 Variance

If c_k is chosen to be $c_k(\mathbf{z}_{-k}) = f_{-k}(\mathbf{z}_{-k})$ then we can again use the approximation (29) for $K \to \infty$ and get the asymptotic variance:

$$\begin{aligned}
\mathrm{Var}[\mathbf{g}] &= \mathrm{Var}\left[\sum_k f_k\mathbf{h}_k\right] && (34)\\
&\approx \mathrm{Var}\left[\sum_k \frac{w_k^2}{2K^2Z^2}\mathbf{h}_k\right] && (35)\\
&= \frac{1}{4K^4Z^4}\sum_k \mathrm{Var}_k\left[w_k^2\mathbf{h}_k\right] && (36)\\
&= \frac{1}{4K^3Z^4}\,\mathrm{Var}_1\left[w_1^2\mathbf{h}_1\right] && (37)\\
&= \mathcal{O}(K^{-3}) && (38)
\end{aligned}$$

where Var_k denotes the variance over the kth approximate posterior $q_\phi(\mathbf{z}_k|\mathbf{x})$, and we used the fact that the latent variables $\{\mathbf{z}_k\}_{k=1}^K$ are i.i.d. and therefore there are no covariance terms.

D Optimal Control for the ESS Limits and Unified Interpolation

D.1 Control Variate for Large ESS

In the gradient estimator $\mathbf{g} = \sum_k \left(\log\hat{Z} - \frac{\partial \log \hat{Z}}{\partial w_k}w_k - c_k\right)\mathbf{h}_k$, we consider the kth term in the sum, where we have that $\hat{Z} - \widetilde{Z}_{[-k]} = \frac{w_k}{K} \to 0$ as $K \to \infty$. We can therefore expand $\log\hat{Z}$ as a Taylor series around $\hat{Z} = \widetilde{Z}_{[-k]}$, obtaining:

$$\log\hat{Z} = \log\widetilde{Z}_{[-k]} + \sum_{p=1}^{\infty}\frac{(-1)^{p+1}}{p}\left(\frac{w_k}{K\widetilde{Z}_{[-k]}}\right)^p \tag{39}$$

$$\frac{\partial \log\hat{Z}}{\partial w_k} = \frac{1}{w_k}\sum_{p=1}^{\infty}(-1)^{p+1}\left(\frac{w_k}{K\widetilde{Z}_{[-k]}}\right)^p . \tag{40}$$

Inserting these results into the gradient estimator and using the expression $\mathbf{g} = \sum_k(f_k + f_{-k} - c_k)\mathbf{h}_k$ we see that

$$\begin{aligned}
f_{-k} &= \log\widetilde{Z}_{[-k]} && (41)\\
f_k &= \sum_{p=1}^{\infty}(-1)^{p+1}\left(\frac{1}{p} - 1\right)\left(\frac{w_k}{K\widetilde{Z}_{[-k]}}\right)^p && (42)\\
&= \sum_{p=2}^{\infty}(-1)^{p}\left(1 - \frac{1}{p}\right)\left(\frac{w_k}{K\widetilde{Z}_{[-k]}}\right)^p . && (43)
\end{aligned}$$

We now use this to simplify the optimal control variate (10) to leading order. Since f_k is order K^{-2}, the term $\mathbb{E}_k\left[f_k\|\mathbf{h}_k\|^2\right]$ will be of order K^{-2} as well. The $l \neq k$ terms $\mathbb{E}_k\left[f_l\mathbf{h}_k^T\mathbf{h}_l\right]$ get non-zero contributions only through the w_k term in f_l. As w_k appears in $\widetilde{Z}_{[-l]}$ with a prefactor K^{-1}, we have $\mathbb{E}_k\left[f_l\mathbf{h}_k^T\mathbf{h}_l\right] = \mathcal{O}(K^{-3})$ for $l \neq k$, and the sum of these terms is $\mathcal{O}(K^{-2})$. Overall, this means that the second term in the control variate only gives a contribution of $\mathcal{O}(K^{-2})$ and thus can be ignored:

$$c_k \approx \log \widetilde{Z}_{[-k]} = \log \frac{1}{K}\sum_{l\neq k} w_l = \log \frac{1}{K-1}\sum_{l\neq k} w_l + \log(1-\frac{1}{K}) \; . \tag{44}$$

Note that in the simplifying approximation in Section 4 we argue that the $l \neq k$ terms $\mathbb{E}_k\left[f_l\mathbf{h}_k^T\mathbf{h}_l\right]$ can be omitted and only the $l = k$ term retained. Here we show that their overall contribution is the same order as the $l = k$ term. These results are not in contradiction because here we are only discussing orders and not the size of terms.

D.2 Control Variate for Small ESS

In the case ESS ≈ 1 we can write $\log \hat{Z}$ as a sum of two terms:

$$\log \hat{Z} = \log \frac{w_{k'}}{K} + \log\left(1 + \frac{K\tilde{Z}_{[-k']}}{w_{k'}}\right) \; , \tag{45}$$

where $w_{k'}$ is the dominating weight. The first term dominates and the second can be ignored to leading order. We will leave out a derivation for non-leading terms for brevity. So the gradient estimator $\mathbf{g} = \sum_k \left(\log \hat{Z} - \frac{\partial \log \hat{Z}}{\partial w_k} w_k - c_k\right)\mathbf{h}_k$ simply becomes $\mathbf{g} \approx \sum_k \left(\log \frac{w_{k'}}{K} - \delta_{k,k'} - c_k\right)\mathbf{h}_k$. This corresponds to $f_k = \delta_{k,k'}\log w_{k'}$ and $f_{-k} = (1-\delta_{k,k'})\log w_{k'} - \delta_{k,k'} - \log K$. Inserting this into Equation (11) we get:

$$\mathbf{g} = \sum_k \left(f_k - \sum_l \frac{\mathbb{E}_k\left[f_l\mathbf{h}_k^T\mathbf{h}_l\right]}{\mathbb{E}_k\left[\|\mathbf{h}_k\|^2\right]}\right)\mathbf{h}_k = \left(\log w_{k'} - \frac{\mathbb{E}_{k'}\left[\log w_{k'}\|\mathbf{h}_{k'}\|^2\right]}{\mathbb{E}_{k'}\left[\|\mathbf{h}_{k'}\|^2\right]}\right)\mathbf{h}_{k'} \; . \tag{46}$$

Estimating the expectation $\mathbb{E}_{k'}[\dots]$ in Equation (46) using i.i.d. samples from $q_\phi(\mathbf{z}|\mathbf{x})$ is computationally involved. Therefore we resort to the approximation $\mathbf{g} \approx \sum_k \left(\log \frac{w_{k'}}{K} - \delta_{k,k'} - c_k\right)\mathbf{h}_k$ and $\delta_{k,k'} \approx v_k$, which holds in the limit ESS $\rightarrow 1$. We get:

$$c_k \approx \log \hat{Z}_{[-k]} - v_k = \log \frac{1}{K-1}\sum_{l\neq k} w_l - v_k \; . \tag{47}$$

Relying on the approximation $\delta_{k,k'} \approx v_k$ corresponds to suppressing the term $-v_k$ of the prefactors $d_k = \log \hat{Z} - v_k$ and does not guarantee the resulting objective to be unbiased for ESS > 1. Suppressing this term has been explored in depth for the pathwise gradient estimator [32]. The gradient estimator $\sum_k v_k\mathbf{h}_k$ corresponds to *wake-phase* update in RWS.

D.3 Unified Interpolation

We unify the two ESS limits under a unifying expression OVIS$_\sim$ defined for a scalar $\gamma \in [0,1]$:

$$c_k^\gamma := \log \hat{Z}_{[-k]} - \gamma v_k + (1-\gamma)\log(1-1/K) \tag{48}$$

where

$$c_k^0 = \log \frac{1}{K-1}\sum_{l\neq k} w_l + \log(1-1/K) \tag{49}$$

$$c_k^1 = \log \frac{1}{K-1}\sum_{l\neq k} w_l - v_k \; . \tag{50}$$

E Rényi Importance Weighted Bound

All the analysis applied to the score function estimator for the importance weighted bound including asymptotic SNR can directly be carried over to the Rényi importance weighted bound $\mathcal{L}_K^\alpha(\mathbf{x})$ because all the independence properties are unchanged. The score function estimator of the gradient of ϕ is given by

$$\nabla_\phi \mathcal{L}_K^\alpha(\mathbf{x}) = \sum_k \left(\frac{1}{1-\alpha} \log \hat{Z}(\alpha) - v_k(\alpha) \right) \mathbf{h}_k, \qquad v_k(\alpha) = \frac{w_k^{1-\alpha}}{\sum_l w_l^{1-\alpha}} \,. \tag{51}$$

The OVIS$_{\mathrm{MC}}$ formulation holds using $d_k = \frac{1}{1-\alpha} \log \hat{Z}(\alpha) - v_k(\alpha)$ within the equation 12. Similarly for the asymptotic expression OVIS$_\sim$, the unified control variate 17 becomes:

$$c_k^\gamma := \log \frac{1}{1-\alpha} \log \hat{Z}_{[-k]}(\alpha) - \gamma v_k + (1-\gamma)\log(1 - 1/K) \tag{52}$$

F Gradient Estimators Review

In this paper, gradient *ascent* is considered (i.e. maximizing the objective function). The expression of the gradient estimators presented below are therefore adapted for this setting.

VIMCO The formulation of the VIMCO [14] control variate exploits the structure of $\hat{Z} := \frac{1}{K}\sum_l w_l$ using $c_k := c_k(\mathbf{z}_{-k}) = \log \frac{1}{K} \sum_{l \neq k} w_l + \hat{w}_{[-k]}$ where $\hat{w}_{[-k]}$ stands for the arithmetic or geometric average of the weights w_l given the set of outer samples $\mathbf{z}_{-k}$. Defining $\log \hat{Z}_{[-k]} := c_k$, the VIMCO estimator of the gradients is

$$\nabla_\phi \mathcal{L}_K = \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})} \left[\underbrace{\sum_{k=1}^{K} \left(\log \hat{Z} - \log \hat{Z}_{[-k]} \right) \mathbf{h}_k}_{\textbf{(a)}} + \underbrace{\sum_{k=1}^{K} v_k \nabla_\phi \log w_k}_{\textbf{(b)}} \right] . \tag{53}$$

We refer to [14] for the derivation. Here, the term $\hat{Z}_{[-k]}$ can be expressed using the arithmetic and the geometric averaging [14]. The leave-one-sample estimate can be expressed as

$$\hat{Z}_{[-k]} = \frac{1}{K} \sum_{l \neq k} w_l + \hat{w}_{[-k]} \text{ with } \begin{cases} \hat{w}_{[-k]} = \frac{1}{K-1} \sum_{l \neq k} w_l & \text{(arithmetic)} \\ \hat{w}_{[-k]} = \exp \frac{1}{K-1} \sum_{l \neq k} \log w_l & \text{(geometric)} \end{cases} \tag{54}$$

The term **(b)** is well-behaved because it is a convex combination of the K gradients $\nabla_\phi \log w_k$. However, the term **(a)** may dominate the term **(b)**. In contrast to VIMCO, OVIS allows controlling the variance of both terms **(a)** and **(b)**, resulting in a more optimal variance reduction. In the Reweighted Wake Sleep (RWS) with wake-wake-ϕ update, the gradient of the parameters ϕ of the inference network corresponds to the negative of the term **(b)**.

Wake-sleep The algorithm [19] relies on two separate learning steps that are alternated during training: the *wake-phase* that updates the parameters of the generative model θ and the *sleep-phase* used to update the parameters of the inference network with parameters ϕ. During the *wake-phase*, the generative model is optimized to maximize the evidence lower bound $\mathcal{L}_1$ given a set of observation $\mathbf{x} \sim p(\mathbf{x})$. During the *sleep-phase*, a set of observations and latent samples are *dreamed* from the model: $\mathbf{x}, \mathbf{z} \sim p_\theta(\mathbf{x}, \mathbf{z})$ and the parameters ϕ of the inference network are optimized to minimize the KL divergence between the true posterior of the generative model and the approximate posterior: $\mathcal{D}_{\mathrm{KL}}\left(p_\theta(\mathbf{z}|\mathbf{x}) || q_\phi(\mathbf{z}|\mathbf{x})\right)$.

Reweighted Wake-Sleep (RWS) extends the original Wake-Sleep algorithm for importance weighted objectives [20]. The generative model is now optimized for the importance weighted bound $\mathcal{L}_K$, which gives the following gradients

$$\nabla_\theta \mathcal{L}_K = \mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})} \left[\sum_k v_k \nabla_\theta \log w_k \right] \quad \text{(wake-phase } \theta) \,. \tag{55}$$

The parameters ϕ of the inference network are optimized given two updates: the *sleep-phase* ϕ an the *wake-phase* ϕ. The *sleep-phase* ϕ is identical to the original Wake-Sleep algorithm, the gradients of the parameters ϕ of the inference model are given by

$$-\nabla_\phi \mathbb{E}_{p_\theta(\mathbf{x})}\left[\mathcal{D}_{\mathrm{KL}}\left(p_\theta(\mathbf{z}_{1:K}|\mathbf{x})||q_\phi(\mathbf{z}_{1:K}|\mathbf{x})\right)\right] = \mathbb{E}_{p_\theta(\mathbf{z}_{1:K},\mathbf{x})}\left[\sum_k \mathbf{h}_k\right] \quad \text{(sleep-phase } \phi) \, . \tag{56}$$

The *wake-phase* ϕ differs from the original Wake-Sleep algorithm that samples $\mathbf{x}, \mathbf{z}$ are sampled respectively from the dataset and from the inference model $q_\phi(\mathbf{z}|\mathbf{x})$. In this cases the gradients are given by:

$$-\nabla_\phi \mathbb{E}_{p(\mathbf{x})}\left[\mathcal{D}_{\mathrm{KL}}\left(p_\theta(\mathbf{z}_{1:K}|\mathbf{x})||q_\phi(\mathbf{z}_{1:K}|\mathbf{x})\right)\right] = \mathbb{E}_{p(\mathbf{x})}\left[\mathbb{E}_{q_\phi(\mathbf{z}_{1:K}|\mathbf{x})}\left[\sum_k v_k \mathbf{h}_k\right]\right] \quad \text{(wake-phase } \phi) \, . \tag{57}$$

Critically, in Variational Autoencoders one optimizes a lower bound of the marginal log-likelihood ($\mathcal{L}_K$), while RWS instead optimizes a biased estimate of the marginal log-likelihood $\log p(\mathbf{x})$. However, the bias decreases with K [20]. [9] shows that RWS is a method of choice for training deep generative models and stochastic control flows. In particular, [9] shows that increasing the budget of particles K benefits the learning of the inference network when using the wake-phase update (Wake-Wake algorithm).

We refer the reader to [20] for the derivations of the gradients and [9] for an extended review of the RWS algorithms for the training of deep generative models.

The Thermodynamic Variational Objective (TVO) The gradient estimator consists of expressing the marginal log-likelihood $\log p_\theta(\mathbf{x})$ using Thermodynamic Integration (TI). Given two unnormalized densities $\tilde{\pi}_0(\mathbf{z})$ and $\tilde{\pi}_1(\mathbf{z})$ and their respective normalizing constants Z_0, Z_1 with $Z_i = \int \tilde{\pi}_i(\mathbf{z})d\mathbf{z}$ given the unnormalized density $\tilde{\pi}_\beta(\mathbf{z}) := \pi_1(\mathbf{z})^\beta \pi_0^{1-\beta}(\mathbf{z})$ parameterized by $\beta \in [0,1]$, and the corresponding normalized density $\pi_\beta(\mathbf{z}) = \tilde{\pi}_\beta(\mathbf{z}) / \int \tilde{\pi}_\beta(\mathbf{z})d\mathbf{z}$, TI seeks to evaluate the ratio of the normalizing constants using the identity

$$\log Z_1 - \log Z_0 = \int_0^1 \mathbb{E}_{\pi_\beta}\left[\frac{d \log \tilde{\pi}_\beta(\mathbf{z})}{d\beta}\right] d\beta \, . \tag{58}$$

[21] connects TI to Variational Inference by setting the base densities as $\tilde{\pi}_0(\mathbf{z}) = q_\phi(\mathbf{z}|\mathbf{x})$ and $\tilde{\pi}_1(\mathbf{z}) = p_\theta(\mathbf{x}, \mathbf{z})$, which gives the Thermodynamic Variational Identity (TVI):

$$\log p_\theta(\mathbf{x}) = \int_0^1 \mathbb{E}_{\pi_\beta}\left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}\right] d\beta. \tag{59}$$

Applying left Riemannian approximation yields the Thermodynamic Variational Objective (TVO):

$$\mathrm{TVO}(\theta, \phi, \mathbf{x}) = \frac{1}{P}\left[\mathrm{ELBO}(\theta, \phi, \mathbf{x}) + \sum_{p=1}^{P-1} \mathbb{E}_{\pi_{\beta_P}}\left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}\right]\right] \leq \log p_\theta(\mathbf{x}) \, . \tag{60}$$

Notably, the integrand $\mathbb{E}_{\pi_\beta}\left[\log \frac{p_\theta(\mathbf{x},\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}\right]$ is monotically increasing, which implies that the TVO is a lower-bound of the marginal log-likelihood.

The TVO allows connecting both Variational Inference and the Wake-Sleep objectives by observing that when using a partition of size $P = 1$, the left Riemannian approximation of the TVI, $\mathrm{TVO}_1^L(\theta, \phi, \mathbf{x}) = \mathrm{ELBO}(\theta, \phi, \mathbf{x})$ and the right Riemannian approximation of the TVI, $\mathrm{TVO}_1^U(\theta, \phi, \mathbf{x})$ is an upper bound of the marginal log-likelihood and equals the objective being maximized in the *wake-phase* for the parameters ϕ of the inference network.

Estimating the gradients of the TVO requires computing the gradient for each of the P expectations $\mathbb{E}_{\pi_{\lambda,\beta}}\left[f_\lambda(\mathbf{z})\right]$ with respect to a parameter $\lambda := \{\theta, \phi\}$ where $f_\lambda(\mathbf{z}) = \log \frac{p_\theta(\mathbf{x},\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}$ and $\mathbf{x}$ is fixed. In the general case, differentiation through the expectation is not trivial. Therefore the authors propose a score function estimator

$$\nabla_\lambda \mathbb{E}_{\pi_{\lambda,\beta}}\left[f_\lambda(\mathbf{z})\right] = \mathbb{E}_{\pi_{\lambda,\beta}}\left[\nabla_\lambda f_\lambda(\mathbf{z})\right] + \mathrm{Cov}_{\pi_{\lambda,\beta}}\left[\nabla_\lambda \log \tilde{\pi}_{\lambda,\beta}(\mathbf{z}), f_\lambda(\mathbf{z})\right] \ , \tag{61}$$

where the covariance term can be expressed as

$$\mathbb{E}_{\pi_{\lambda,\beta}}\left[\left(f_\lambda(\mathbf{z}) - \mathbb{E}_{\pi_{\lambda,\beta}}\left[f_\lambda(\mathbf{z})\right]\right)\left(\nabla_\lambda \log \tilde{\pi}_{\lambda,\beta}(\mathbf{z}) - \mathbb{E}_{\pi_{\lambda,\beta}}\left[\nabla_\lambda \log \tilde{\pi}_{\lambda,\beta}(\mathbf{z})\right]\right)\right] \ . \tag{62}$$

The covariance term arises when differentiating an expectation taken over a distribution with an intractable normalizing constant, such as $\pi_\beta(\mathbf{z})$ in the TVO. The normalizing constant can be substituted out, resulting in a covariance term involving the tractable un-normalized density $\tilde{\pi}_\beta(\mathbf{z})$. Hence, such a covariance term does not usually arise in IWAE due to the derivative of $q_\phi(\mathbf{z}|\mathbf{x})$ being available in closed form.

G Gaussian Model

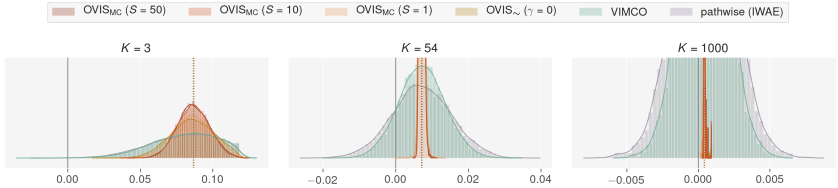


Figure 5: Distribution of the gradients for an arbitrarily chosen component of the parameter $\mathbf{b}$. The tight control of the variance provided by OVIS allows keeping the distribution of gradients off-center.

Distribution of gradients We report the distributions of the 10^4 MC estimates of the gradient of the first component b_0 of the parameter $\mathbf{b}$. Figure 5. The pathwise estimator and VIMCO yield estimates which distributions are progressively centered around zero as $K \to \infty$. The faster decrease of the variance of the gradient estimate for OVIS results in a distribution of gradients that remains off-centered.

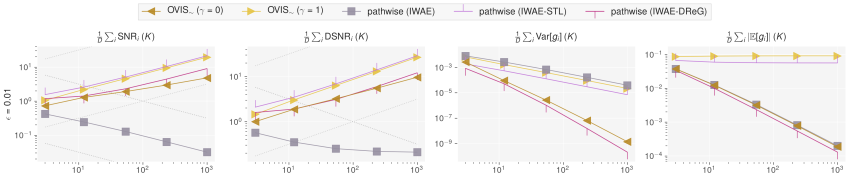


Figure 6: Asymptotic analysis of the gradients for OVIS$_\sim$ and the STL and DReG IWAE estimators.

Analysis for advanced pathwise IWAE estimators We perform the experiment 3 using additional pathwise estimators: STL [32] and DReG-IWAE [24]. Both the STL and OVIS$_\sim(\gamma = 1)$ rely on the suppression of the term $\sum_k v_k \mathbf{h}_k$ from the gradient estimate and adopt the same behaviour: the variance decreases at a slower rate than OVIS$_\sim(\gamma = 0)$ and DReG, however, its bias remains constant as K is increased.

Fitting the Gaussian Model

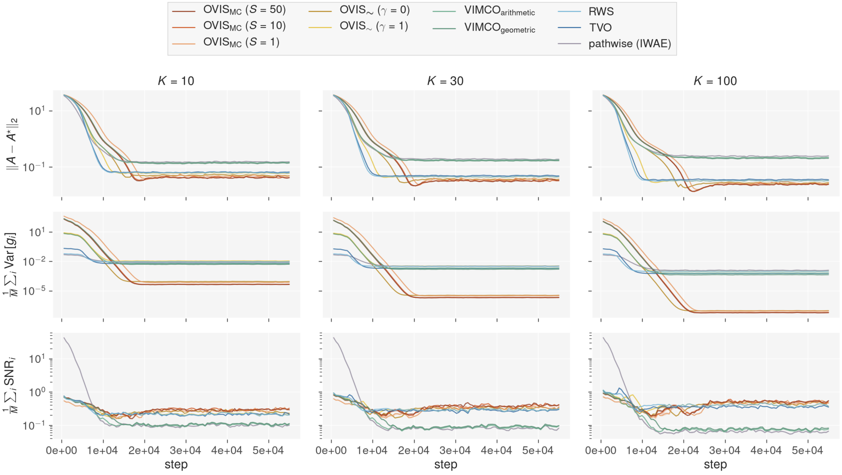


Figure 7: Fitting the Gaussian toy model from section 6.1 and measuring the $\mathcal{L}_2$ distance with the optimal parameters as well as the variance and the SNR of the gradient estimates. OVIS methods target the optimal parameters $A^\star$ of the inference network more accurately than the baseline methods.

We study the relative effect of the different estimators when training the Gaussian toy model from section 6.1. The model is trained for 5.000 epochs using the Adam optimizer with a base learning rate of 10^{-3} and with a batch-size of 100. In Figure 7, we report the L_2 distance from the model parameters A to the optimal parameters $A^\star$, the parameters-average SNR and parameters-average variance of the inference network ($\phi = \{A, \mathbf{b}\}, M = \mathrm{card}(\phi)$). We compare OVIS methods with VIMCO, the pathwise IWAE, RWS and the TVO for which we picked a partition size $P = 5$ and $\beta_1 = 10^{-3}$, although no extensive grid search has been implemented to identify the optimal choice for this parameters.

OVIS yields gradient estimates of lower variance than the other methods. The inference network solutions given by OVIS are slightly more accurate than the baseline methods RWS and the TVO, despite being slower to converge. OVIS, RWS and the TVO exhibit gradients with comparable SNR values, which indicate OVIS yield estimate of lower expected value, thus leading to a smaller maximum optimization step-size. Setting $\gamma = 0$ for OVIS$_\sim$ results in more accurate solutions than using $\gamma = 1$, this coincides with the measured ESS $\approx K$.

H Gaussian Mixture Model

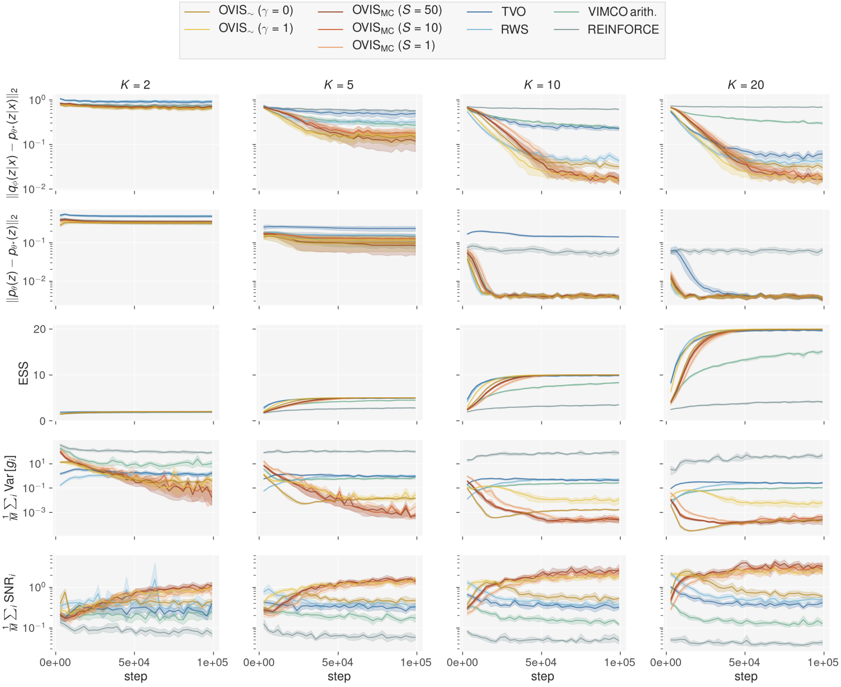


Figure 8: Training curves for the Gaussian Mixture Model for different numbers of particles $K = [2, 5, 10, 20]$ samples averaged over 5 random seeds. The SNR is measured on one mini-batch and averaged over the M parameters of the inference network. In contrast to VIMCO, OVIS estimators all generate gradients with a higher SNR. This results in a more accurate estimate of the true posterior, when compared to VIMCO and the baselines RWS and the TVO.

I Comparison of $\mathrm{OVIS}_\sim$ and $\mathrm{OVIS}_{\mathrm{MC}}$ with under a fixed Particle Budget

$\mathrm{OVIS}_{\mathrm{MC}}$ has complexity requires $K + S$ importance weights whereas $\mathrm{OVIS}_\sim$ requires only K. Estimating ϕ using $\mathrm{OVIS}_{\mathrm{MC}}$ requires a budget of $K' = K + S$ particles. The ratio S/K is a trade-off between the tightness of the bound $\mathcal{L}_K$ and the variance of the control variate estimate. In the main text, we focus on studying the sole effect of the control variate given the bound $\mathcal{L}_K$. This corresponds to a sub-optimal use of the budget K' because $\mathcal{L}_{K'}$ is tighter than $\mathcal{L}_K$. By contrast with the previous experiments, we trained the Gaussian VAE using the budget K' optimally (i.e. relying on $\mathcal{L}_{K'}$ whenever no auxiliary samples are used). We observed that $\mathrm{OVIS}_\sim(\gamma = 1)$ outperforms $\mathrm{OVIS}_{\mathrm{MC}}$ despite the generative model is evaluated using $\mathcal{L}_{K'}$ in all cases (figure 9). This experiment will be detailed in the Appendix.

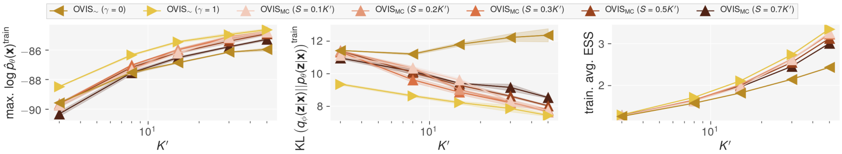


Figure 9: Training the Gaussian VAE model with a fixed and optimally used particle budget $K' = K + S$ and $\alpha = 0.7$.

J Training Curves for the Deep Generative Models

J.1 Sigmoid Belief Network

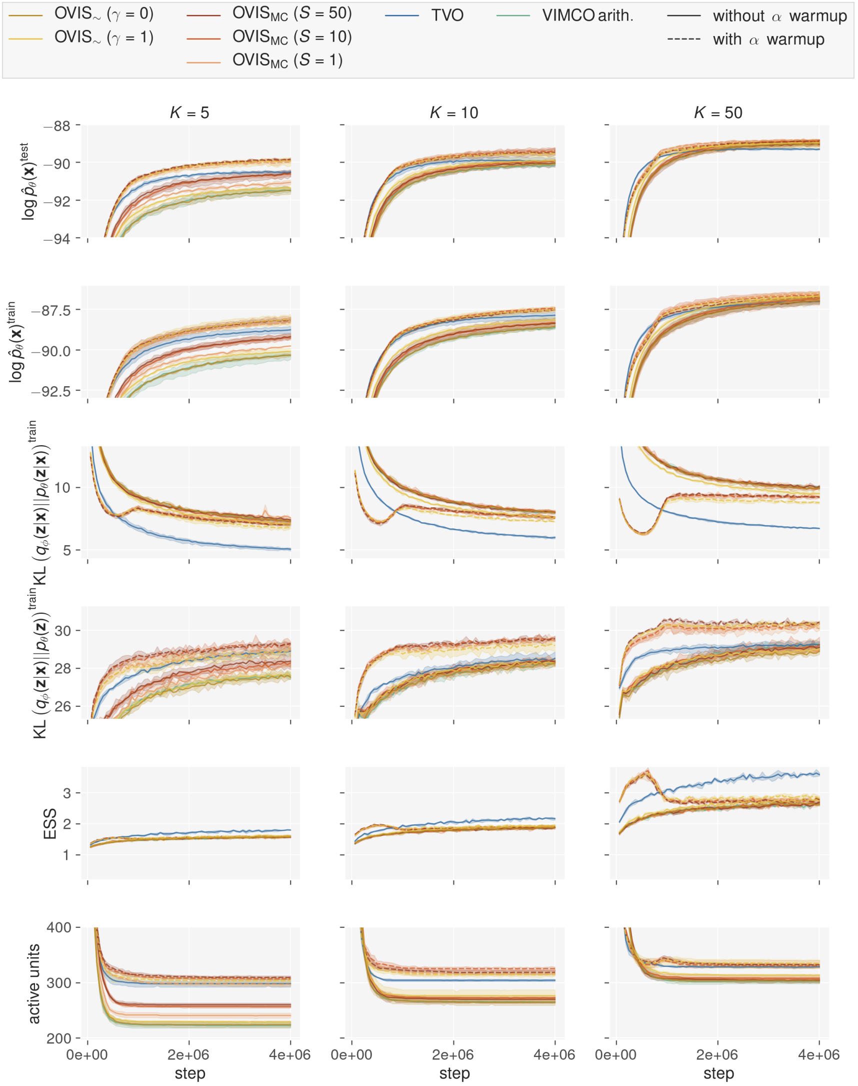


Figure 10: Training curves for the Sigmoid Belief Network using $K = [5, 10, 50]$ particles, using two initial random seeds, with and without using the IWR bound. The number of active units is evaluated as $\text{AU} = \sum_{d=1}^{D} \mathbb{1}\left\{\text{Cov}_{p(\mathbf{x})}\left(\mathbb{E}_{q_\phi(\boldsymbol{z}|\mathbf{x})}\left[\boldsymbol{z}_d\right]\right) \geq 0.01\right\}$ [22] using 1000 MC samples for each element of a randomly sampled subset of 1000 data points. Warming up the model by optimizing for the IWR bound with $\alpha > 0$ allows activating a larger number of units and results in models scoring higher training likelihoods.

J.2 Gaussian Variational Autoencoder

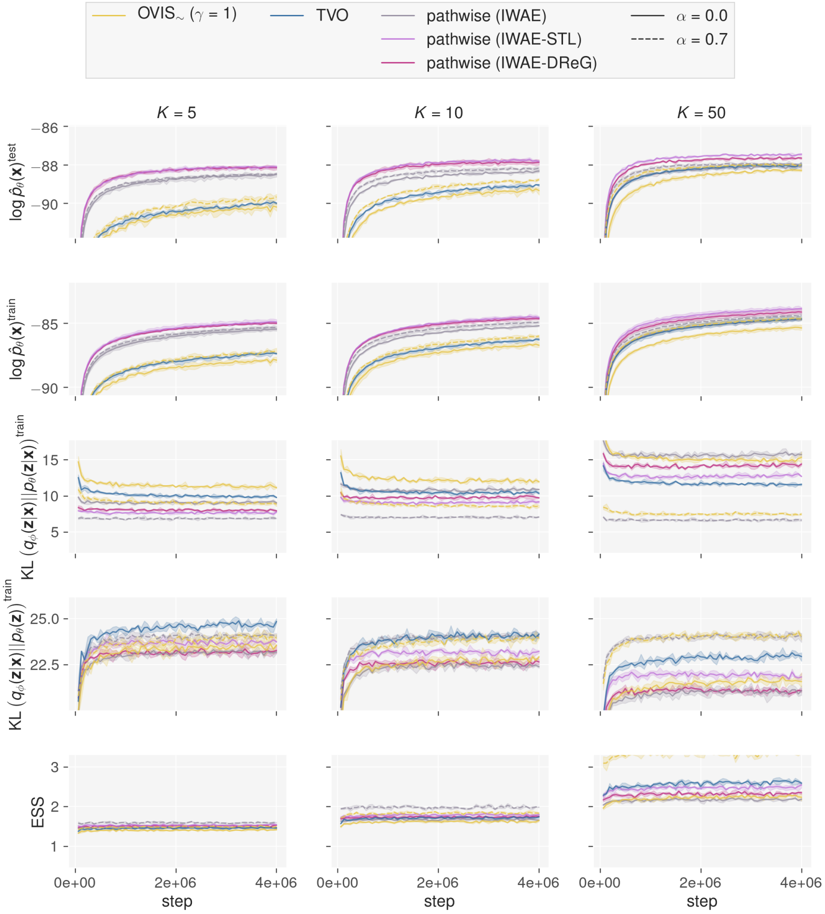


Figure 11: Training curves a Gaussian VAE using $K = [5, 10, 50]$ particles and using two initial random seeds. The OVIS estimators are used in tandem with the IWR bound with α fixed to 0.3. OVIS for the IWR bound yields high-quality inference networks, as measured by the divergence $\mathcal{D}_{\mathrm{KL}}\left(p_\theta(\mathbf{z}|\mathbf{x})||q_\phi(\mathbf{z}|\mathbf{x})\right)$.

K Implementation Details for $\mathrm{OVIS}_{\sim}$

In order to save computational resources for large K values, we implement the following factorization

$$\log \hat{Z} - \log \hat{Z}_{[-k]} = \log \frac{1 - 1/K}{1 - v_k} . \tag{63}$$

In order to guarantee computational stability, we clip the normalized importance weights v_k using the default PyTorch value $\epsilon = 1.19e^{-7}$. The resulting gradient estimate, used in the main experiments, is

$$\mathbf{g} := \sum_k \left(\log \frac{1 - 1/K}{1 - \min(1 - \epsilon, v_k)} + (\gamma - 1) v_k - (1 - \gamma) \log(1 - 1/K) \right) \mathbf{h}_k . \tag{64}$$

Clipping the normalized importance weights can be interpreted as an instance of truncated importance sampling. Hence, the value of ϵ must be carefully selected. In the figure 12, we present a comparison of $\mathrm{OVIS}_{\sim}$ with and without clipping. The experiments indicate that the difference is insignificant when using the default ϵ.

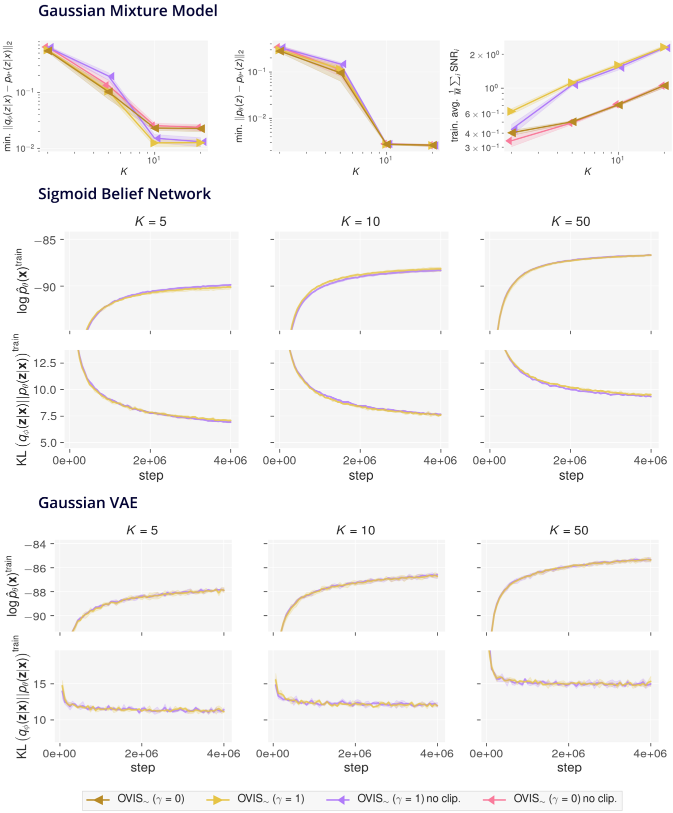


Figure 12: Effect of the importance weight clipping. Training the Gaussian Mixture Model, Sigmoid Belief Network and Gaussian VAE with and without clipping.

APPENDIX C

Can Large Language Models Reason about Medical Questions?

CAN LARGE LANGUAGE MODELS REASON ABOUT MEDICAL QUESTIONS?

Valentin Liévin [1,2] **Christoffer Egeberg Hother**[3] **Ole Winther**[1, 2, 4, 5]
[1] Section for Cognitive Systems, Technical University of Denmark, Denmark
[2] FindZebra, Denmark
[3] Department of Clinical Immunology, Rigshospitalet, Copenhagen University Hospital, Denmark
[4] Center for Genomic Medicine, Rigshospitalet, Copenhagen University Hospital, Denmark
[5] Bioinformatics Centre, Department of Biology, University of Copenhagen, Denmark
`valv@dtu.dk, christoffer.egeberg.hother@regionh.dk, olwi@dtu.dk`

ABSTRACT

Although large language models (LLMs) often produce impressive outputs, they also fail to reason and be factual. We set out to investigate how these limitations affect the LLM's ability to answer and reason about difficult real-world based questions. We applied the human-aligned GPT-3 (InstructGPT) to answer multiple-choice medical exam questions (USMLE and MedMCQA) and medical research questions (PubMedQA). We investigated Chain-of-thought (think step by step) prompts, grounding (augmenting the prompt with search results) and few-shot (prepending the question with question-answer exemplars). For a subset of the USMLE questions, a medical domain expert reviewed and annotated the model's reasoning. Overall, GPT-3 achieved a substantial improvement in state-of-the-art machine learning performance. We observed that GPT-3 is often knowledgeable and can reason about medical questions. GPT-3, when confronted with a question it cannot answer, will still attempt to answer, often resulting in a biased predictive distribution. LLMs are not on par with human performance but our results suggest the emergence of reasoning patterns that are compatible with medical problem-solving. We speculate that scaling model and data, enhancing prompt alignment and allowing for better contextualization of the completions will be sufficient for LLMs to reach human-level performance on this type of task.

1 INTRODUCTION

Self-supervised pre-training promises to leverage the vast quantity of unlabelled data (text, images, audio or videos) to learn general-purpose models that can then be applied to a myriad of downstream problems. Language representations have transformed the field of natural language processing, from simple word vectors (Mikolov et al., 2013; Pennington et al., 2014) to deep contextualized representations (Peters et al., 2018; Vaswani et al., 2017; Devlin et al., 2018; Radford et al., 2018), language models are now ubiquitous in natural language processing, notably, thanks to the success of the Transformer architecture Vaswani et al. (2017) and its high compatibility with massively parallel computation hardware.

In recent years, tremendous resources have been allocated to scale Transformer-based language models Brown et al. (2020); Rae et al. (2021); Chowdhery et al. (2022); Thoppilan et al. (2022); Hoffmann et al. (2022); Smith et al. (2022); Zhang et al. (2022); Lieber et al. (2021); Fedus et al. (2021); Laurençon et al. (2022) to using hundreds of billions of parameters and to training on gigabytes of text. This translated in sustained improvements as measured by a wide variety of benchmarks Srivastava et al. (2022). Large language models (LLMs) are expensive to train and require the development of safeguards before being deployed into real-world systems. Notably, LLMs tend to amplify the social biases present in the training data, have severe and easily exposable deficiencies in their reasoning capabilities, make things up based upon the information stored in their parameters and there is no way to get pointers back to the source data they use to generate their output (Bender et al., 2021). Therefore, deploying LLMs into sensitive areas such as healthcare must be operated with great care Korngiebel & Mooney (2021); Sezgin et al. (2022). Nonetheless, large language models are powerful tools and therefore have the potential to transform the field of machine intelligence.

Prompt-based learning Liu et al. (2021) opens a new way to interact with language models. Whereas the fine-tuning paradigm requires refining the model's parameters, prompt-based learning consists in querying language models with a prompt that contains in-domain examples and/or a task description. In other words, prompt-based learning relies on natural language as a medium for learning to solve new problems. Wei et al. (2022); Kojima et al. (2022) showed that prompts could be designed to trigger LLMs to solve problems using step-by-step reasoning (chain-of-thought). This appeared to be effective at solving problems requiring logic, including mathematical problems Lewkowycz et al. (2022), which were thought to be the weakness of deep language models Rae et al. (2021). Nonetheless, reasoning tasks remain overall challenging for LLMs Srivastava et al. (2022), especially some reasoning-heavy tasks like planning which are out of reach for these models Valmeekam et al. (2022).

Kojima et al. (2022) showed that the prompt "*Let's think step by step*" could trigger GPT-3 to generate multi-step reasoning. Largely inspired by this work, we study whether this technique can be applied to solve medical problems, which require combining multi-steps reasoning, strong natural language understanding capabilities and a high degree of medical knowledge. Applying LLMs to the biomedical domain remains mostly an uncharted area, and early research indicates that prompt-based learning is not yet competitive with other domain and task-specific BERT models Moradi et al. (2021).

The main contributions of this paper are:

- A study of GPT-3's performance with zero-shot chain-of-thought (CoT) prompting on multiple-choice medical board exam question datasets (USMLE and MedMCQA) and a medical reading comprehension dataset (PubMedQA).
- Showing that GPT-3 achieves state-of-the-art performance on the three datasets but is still below human expert performance,
- Providing a medical expert evaluation of a small set of the generated chain of thoughts. The expert review supports that GPT-3 in many cases can reason and exploit memorized expert knowledge.
- Showing that grounding – augmenting the prompt with retrieved information – can be used to improve performances.
- Showing that few-shot prompt-based learning can be used to answer medical questions better, with and without chain-of-thought prompting.

2 BACKGROUND

Autoregressive models learn a left-to-right factorization of the data, they excel in modelling a wide range of data modalities: images Salimans et al. (2017), audio Oord et al., videos Weissenborn et al. (2019) and language Grave et al. (2016); Dai et al. (2019); Shoeybi et al. (2019). In particular for text, autoregressive language modelling has proven to be a good objective for learning general-purpose models that can be applied to solve a wide range of downstream tasks Radford et al. (2018; 2019); Brown et al. (2020). Scaling language models has so far proven to be a successful strategy for improving language models Brown et al. (2020); Rae et al. (2021); Chowdhery et al. (2022); Thoppilan et al. (2022); Srivastava et al. (2022).

There is hope that deep language models can acquire more than superficial statistical dependencies between words: with scale, language models might learn meaningful language representations Li et al. (2021) and might learn problem-solving skills such as common-sense, arithmetic, symbolic and logical reasoning, planning and more. The BIG-Bench Srivastava et al. (2022) is a recent community-wide initiative that aims at quantifying the large gap that remains between human and LLM language capabilities.

Few-shot learning *Pre-train, prompt and predict* (Liu et al., 2021) is an emerging paradigm for applying LLMs to new problems, without fine-tuning the weights. Prompt-based learning consists

Table 1: Prompt templates. In the table below, we use `typewriter style` and brackets to represent `[provided data]` such as the question, additional context, or the answer and `<completions>` generated by GPT-3. We use the pipe symbol `OR` to indicate when a variable can be either `[provided data] OR` a `<completion>`. The symbol $\varnothing$ is used to represent empty strings.

	Zero-shot $\varnothing$	**Zero-shot CoT**
Question	Question: `[Question]`	Question: `[Question]`
CoT	$\varnothing$	Answer: Let's think step by step `<CoT>`
Answer	Answer: among A through D, the answer is `<answer>`	Therefore, among A through D, the answer is `<answer>`
	Zero-shot $\varnothing$ + grounding	**One-shot CoT**
Shot	$\varnothing$	Question: `[Question]`
	$\varnothing$	Answer: Let's think step by step `[Explanation] OR <CoT>`
	$\varnothing$	Therefore, among A through D, the answer is `[answer]`
Context	Context: `[Context]`	$\varnothing$
Question	Question: `[Question]`	Question: `[Question]`
CoT	$\varnothing$	Answer: Let's think step by step `<CoT>`
Answer	Answer: among A through D, the answer is `<answer>`	Therefore, among A through D, the answer is `<answer>`

in augmenting the problem with instructions such that the model's completion of the prompt will correspond to a solution. This allows for LLMs to learn from examples – or *shots* – which are simply incorporated into the prompt. This few-shot learning technique has proven to be highly effective on a wide range of tasks (Brown et al., 2020).

Nonetheless, zero-shot and few-shot learning might not always be sufficient to bridge the gap with humans' expectations, and further finetuning might not be avoidable. As an answer to this limitation, Ouyang et al. (2022) introduced InstructGPT: a GPT-3 model finetuned using reinforcement learning with human feedback to "follow the user's instructions helpfully and safely". InstructGPT outperforms GPT-3 in diverse prompt-based learning scenarios.

Chain-of-Thoughts Wei et al. (2022) showed that LLMs could be applied to system 2 problems by prompting the model to break the problem into reasoning that breaks into multiple steps, coined "chain-of-thought" (CoT). This requires crafting prompts that trigger CoTs as completions for each problem, which makes it difficult to apply to all datasets.

Kojima et al. (2022) demonstrated that CoTs can be generated using a single and domain-agnostic prompt "Let's think step by step" . The CoTs that resulted from that prompt not only appeared to expose valid reasoning but also translated into dramatic improvements, as measured across multiple benchmarks in a zero-shot setting.

Multiple CoTs can be sampled from the generative LLMs, which allows for potentially exploring multiple hypotheses for a single problem. The multiple CoTs can be marginalized, or combined into an ensemble of models. Wang et al. (2022); Li et al. (2022) showed that majority voting or learned heuristics can be used to outperform single-sample CoT methods.

Grounding Large language models memorise part of the knowledge embedded into the training data, nonetheless, models might fail to re-use this knowledge effectively during prediction. Conditioning the predictions on a knowledge base is an alternative research direction for improving language models (Lewis et al., 2020; Borgeaud et al., 2021; Lazaridou et al., 2022). Retrieving documents as part of solving downstream tasks corresponds to the *open-domain* setting in which the model has access to all available knowledge Chen et al. (2017). Given a good information retrieval system and a comprehensive knowledge base, a language model might solve downstream by relying only on its reading comprehension capabilities.

3 PROMPT DESIGN FOR MEDICAL QUESTION ANSWERING

Prompting GPT-3 to answer multiple choice questions We studied two classes of prompts: the CoT-free prompt, denoted $\varnothing$ and a set of five zero-shot CoT prompts (Kojima et al., 2022). In the case of the $\varnothing$ prompt, we queried GPT-3 with a single prompt composed of the context, if any, and the question followed by the extractive prompt which completion is the answer. When applying the zero-shot CoT framework, we utilized ta two-steps prompting scheme with first a reasoning prompt

Table 2: Selected CoT prompts. The prompt $\varnothing$ corresponds to the CoT-free setting.

0 – $\varnothing$
1 – *Let's think step by step*
2 – *Let's think step by step like a medical expert*
3 – *Let's use step by step inductive reasoning, given the medical nature of the question*
4 – *Let's differentiate using step by step reasoning like a medical expert*
5 – *Let's derive the differential diagnosis*

which completion is the CoT, and second and extractive prompt which completion is the answer. All prompt designs are summarized in Table 1.

We researched whether domain-specific CoT prompts could be helpful. We tested the original prompt from Kojima et al. (2022) ("*Let's think step by step*"), a medical version of this prompt ("*Let's think step by step like a medical expert*") and the top three best performing prompts among an initial selection of 30 CoT prompts. The prompts were tested using a set of 100 exam questions taken from the US Medical Licensing Examination (USMLE). The validation study can be found in Appendix A. The selected prompts, including the CoT-free prompt, are summarized in Table 2.

Grounding We investigated whether grounding the model with further context could improve the answering accuracy. We experimented with a simple BM25 retriever and used Wikipedia as a knowledge base. The articles were converted into overlapping passages of size 100 words and indexed along with the titles. Given a question $\mathbf{q}$, an answer choice $\mathbf{a}$, and weights $\beta_1 = 1, \beta_2 = 1, \beta_3 = 0.5$, we retrieved passages based on a composite BM25 score defined as

$$\mathrm{score}(\mathbf{q}, \mathbf{a}) = \beta_1 \cdot \mathrm{BM25}(\mathbf{q}, \textit{content}) + \beta_2 \cdot \mathrm{BM25}(\mathbf{a}, \textit{content}) + \beta_3 \cdot \mathrm{BM25}(\mathbf{a}, \textit{title}) \ . \tag{1}$$

For all the grounded experiments, except when dealing with the PubMedQA dataset, for which we use the provided context, we prepend the question with the top-1 retrieved passage for each answer choice.

Few-shot learning We experimented with inserting examplars (or *shots*) of question-answer pairs in the prompts. We built each shot using the same template as for the zero-shot setting, except that we replaced the answer prediction with the true answer. Additionaly, When using CoT prompting, we either generated a CoT for each shot, replaced the CoT with the question explanation (when provided), or omitted the CoT and the accompanying CoT prompt (results in Table 9). The one-shot CoT prompt design is presented in Table 1.

4 EXPERIMENTS

We first introduce the three datasets, report the answering accuracy on all datasets and discuss the success and failure patterns. Finally, we show that few-short learning could be used to improve performances further. The code repository and samples of geenrated CoTs for each dataset can be accessed at vlievin.github.io/medical-reasoning.

4.1 DATASETS

We focus the study on three medical multiple-choice question answering datasets: USMLE, MedMCQA and PubMedQA. The three datasets are summarized in Table 3. The USMLE and MedMCQA datasets require retrieving further context from medical books or Wikipedia, which corresponds to the *open-domain* setting (Chen et al., 2017). Both the MedMCQA and the PubMedQA datasets come with detailed explanations for each question (long answer).

USMLE The USMLE (MedQA) dataset Jin et al. (2021) gathers historical questions from the United States Medical Licensing Examination (USMLE), which targets trained medical professionals. The questions are notorious for being challenging as they often require strong problem-solving skills coupled with comprehensive medical knowledge. Each question features a description of a medical case and a question that emulates the real clinical setting.

Table 3: Summary of the medical question answering datasets.

	USMLE	**MedMCQA**	**PubMedQA**
# questions (train/valid./test)	10.2k/1.3k/1.3k	182.8k/4.2k/6.1k	450/50/500
# words / question	116.6	12.7	14.4
Answer options	A/B/C/D	A/B/C/D	yes/no/maybe
Explanation length	–	66.2	43.2
Context length	–	–	238.9
Data source	National Medical Board Examination (US)	AIIMS and NEET PG entrance exams	expert-annotated PubMed abstracts

MedMCQA The MedMCQA (Pal et al., 2022) is a large-scale multiple-choice question answering collected from Indian medical school entrance exams (AIIMS and NEET-PG). The MedMCQA covers a broad range of medical topics (dentistry, psychiatry, surgery, ...) and require being able to follow a variety of reasoning types (logic, factual, comparison, ...).[6]

PubMedQA The PubMedQA dataset (Jin et al., 2019) is a collection of expert-annotated yes/no/maybe research questions derived from PubMed abstracts. Whereas the questions from the USMLE and the MedMCQA datasets are self-contained and might be answered using general medical knowledge and methodology, each PubMedQA question is contextualized on a provided abstract. Therefore PubMedQA primalry focuses on evaluating reading comprehension skills.

4.2 ZERO-SHOT BENCHMARK

Table 4: Answering accuracy of GPT-3 on the USMLE dataset in a zero-shot setting.

model	**grounding**	**CoT prompt**	**accuracy (%)**	**CoT length**
GPT-3	✗	0 – ∅	46.0	0
GPT-3	✗	1 – *Let's think step by step*	47.1	129
GPT-3	✗	2 – *Let's think step by step like a medical expert*	46.8	173
GPT-3	✗	3 – *Let's use step by step inductive reasoning, (...)*	46.0	171
GPT-3	✗	4 – *Let's differentiate using step by step reasoning (...)*	45.6	207
GPT-3	✗	5 – *Let's derive the differential diagnosis step by step*	45.1	199
GPT-3	✓	0 – ∅	47.3	0
GPT-3	✓	1 – *Let's think step by step*	45.9	141
GPT-3	✓	2 – *Let's think step by step like a medical expert*	47.0	172
GPT-3	✓	3 – *Let's use step by step inductive reasoning, (...)*	45.6	165
GPT-3	✓	4 – *Let's differentiate using step by step reasoning (...)*	45.9	196
GPT-3	✓	5 – *Let's derive the differential diagnosis step by step*	47.4	195
GPT-3	✗	0+1+2+3+4+5 (majority voting, n=6)	50.0	–
GPT-3	✓	0+1+2+3+4+5 (majority voting, n=6)	49.3	–
GPT-3	✗ + ✓	0+1+2+3+4+5 (majority voting, n=12)	**53.1**	–
BioLinkBERT[1]	✓	–	44.6	–
Human (passing score)[2]	✓	–	$\geqslant 60$	–

We applied the largest human-aligned GPT-3 (InstructGPT, `text-davinci-002`, Ouyang et al. (2022), 175B parameters) to answering medical questions in a zero-shot setting, with and without chain-of-thought prompting, with and without grounding. We sampled one completion per prompt with a temperature of zero and limited the completions to a maximum length of 1024 tokens. The predicted answers were extracted from the completions following the method described in Kojima et al. (2022).

---

[1] Results and model from Yasunaga et al. (2022)

[2] https://www.usmle.org/scores-transcripts

[3] Results from Pal et al. (2022), model from Gu et al. (2021)

[4] AIIMS: https://collegedunia.com/exams/aiims-mbbs/cutoff

[5] NEET PG: https://medicine.careers360.com/articles/neet-pg-cut-off

[6] In this version of the paper, we estimated the MedMCQA performances based on a subset of 1k validation samples to limit the overall running costs

Table 5: Answering accuracy of GPT-3 on the MedMCQA dataset in a zero-shot setting. GPT-3 is evaluated on a subset of randomly sampled 1k validation questions whereas the baselines are evaluated on the whole set. The accuracy is reported for the validation and test splits (reported as `validation` / `test`).

model	grounding	CoT prompt	accuracy (%)	CoT length
GPT-3	✗	0 – ∅	44.0 / –	0
GPT-3	✗	1 – *Let's think step by step*	40.8 / –	140
GPT-3	✗	2 – *Let's think step by step like a medical expert*	43.3 / –	143
GPT-3	✗	3 – *Let's use step by step inductive reasoning, (...)*	38.8 / –	152
GPT-3	✗	4 – *Let's differentiate using step by step reasoning (...)*	37.1 / –	156
GPT-3	✗	5 – *Let's derive the differential diagnosis step by step*	42.1 / –	137
GPT-3	✓	0 – ∅	46.7 / –	0
GPT-3	✓	1 – *Let's think step by step*	42.2 / –	149
GPT-3	✓	2 – *Let's think step by step like a medical expert*	45.8 / –	170
GPT-3	✓	3 – *Let's use step by step inductive reasoning, (...)*	41.6 / –	175
GPT-3	✓	4 – *Let's differentiate using step by step reasoning (...)*	41.3 / –	185
GPT-3	✓	5 – *Let's derive the differential diagnosis step by step*	41.8 / –	176
GPT-3	✗	0+1+2+3+4+5 (majority voting, n=6)	42.4 / –	–
GPT-3	✓	0+1+2+3+4+5 (majority voting, n=6)	**48.8 / –**	–
GPT-3	✗ + ✓	0+1+2+3+4+5 (majority voting, n=12)	47.6 / –	–
PubMedBERT[3]	✗	–	40 / 41	–
PubMedBERT[3]	✓	–	43 / 47	–
Human (merit candidate) [3]	✓	–	⩾90.0	–
Human (passing score) [45]	✓	–	⩾50.0	–

Table 6: Answering accuracy of GPT-3 on the PubMedQA dataset in a zero-shot setting.

model	grounding	CoT prompt	accuracy (%)	CoT length
GPT-3	✓	0 – ∅	**73.2**	0
GPT-3	✓	1 – *Let's think step by step*	60.0	170
GPT-3	✓	2 – *Let's think step by step like a medical expert*	59.8	221
GPT-3	✓	3 – *Let's use step by step inductive reasoning, (...)*	66.2	170
GPT-3	✓	4 – *Let's differentiate using step by step reasoning (...)*	58.0	218
GPT-3	✓	5 – *Let's derive the differential diagnosis step by step*	55.6	205
GPT-3	✓	0+1+2+3+4+5 (majority voting, n=6)	70.4	–
BioLinkBERT large[3]	✓	–	72.2	–
Human	✓	–	78.0	–

Following Wang et al. (2022), we report performances using an ensemble of all the prompts using majority voting. We report the human baselines and the current state-of-the-art BERT-based methods. We gathered all results, including the average number of generated tokens per chain-of-thought in 4 for the USMLE dataset, in Table 5 for the MedMCQA dataset and in Table 6 for the PubMedQA dataset.

Without CoT GPT-3 outperformed the domain-specific and finetuned BERT-based baselines on the three datasets, despite not being explicitly trained to answer such questions. GPT-3 outperformed the grounded BERT baselines on the USMLE exam questions (Table 4, 46.0% for GPT-3, 44.6% for BERT), on the MedMCQA exam questions (Table 5, 44.0% for GPT-3, 43.0% for BERT) and on the PubMedQA contextualized questions (Table 6, 73.2% for GPT-3, 72.2% for BERT).

With CoT Zero-shot CoT-free prompting remained a better alternative to zero-shot CoT prompting. Performances were lower for each of the CoT prompts compared to the CoT-free prompt, except for the USMLE dataset for which half of the CoT prompts resulted in small improvements over the CoT-free prompt ∅ (47.1% accuracy using prompt number 1, 46.0% using prompt ∅).

Ensembling and Grounding The ensemble of prompts overall outperformed the single prompts (50% on the USMLE dataset without grounding, 48.8% on the MedMCQA dataset with grounding). The PubMedQA dataset is an exception: the ensemble of prompts did not outperform the single CoT-free prompt ∅, which performed exceptionally well with an accuracy of 73.2%.

In an attempt to exploit the good reading comprehension skills of GPT-3, we conditioned the completions of USMLE and MedMCQA questions on Wikipedia passages. When using the $\varnothing$ prompt, we recorded minor gains on the USMLE (+1.3%) and on the MedMCQA (+2.6%) datasetfoos. However, using the ensemble of the six prompts, grounding marginally improved accuracy on the MedMCQA dataset (+6.47%: 42.4% without grounding, 48.8% with grounding).

In the case of the USMLE dataset, we found that the grounding-free predictions still had to be included in the ensemble to reach superior performances, which resulted in an outstanding accuracy of 53.1% (12 predictions per question, six prompts, with and without grounding).

4.3 ANALYSIS OF THE SUCCESS AND FAILURE OF THE COTS

Table 7: Frequency of observed patterns (A, B, C, D, E and F) identified among 50 CoT based on USMLE questions. The table is split based on the predicted answer (correct/incorrect). A CoT is labelled as containing a pattern if it contains at least one observable sign.

	Pattern	**Correct answers (16)**	**Incorrect answers (34)**	**Total (50)**
A	Correct reasoning step	94% (15)	59% (20)	70% (35)
B	Correct recall of knowledge	87% (14)	65% (22)	72% (36)
C	Correct reading comprehension	100% (16)	85% (29)	90% (45)
D	Incorrect reasoning step	12% (2)	86% (29)	62% (31)
E	Incorrect or insufficient knowledge	25% (4)	74% (25)	58% (29)
F	Incorrect reading comprehension	6% (1)	50% (17)	36% (18)

We considered three general skills that we expect are required to be mastered to answer medical questions: (1) performing non-trivial reasoning steps, (2) recalling knowledge that is not provided in the context and (3) ability to comprehend the question and the context. Based on the three skills, we defined three success patterns (A, B, C) and three failure patterns (D, E, F).

A subset of 50 CoTs generated based on USMLE questions were annotated by a medical expert (CEH) using the six categories. For each category and each CoT, we reported a match if the pattern could be observed. This means that a CoT can be labelled with both a correct and an incorrect pattern for the same skill. We showcase three chain-of-thoughts (three in Table 10 with patterns highlighted in the text whenever possible. 27 additional annotations are presented in Appendix C).

We report the frequencies of occurrence for the six patterns in Table 7. We found that most of the questions answered incorrectly triggered generating CoTs that contained reasoning errors (pattern D, 86%), and that showed a lack of knowledge (pattern E, 74%). Misunderstanding of the questions or the context was less frequently observed (Pattern F, 50%). We observed that CoTs leading to questions answered correctly could exhibit failure patterns but we also observed that the CoTs leading to incorrect answers were not entirely incorrect, as 59% contained at least one correct reasoning step, 65% of showed proper recall of knowledge. Furthermore, we inspected the CoTs leading to wrong answers and found that 47% of those were inconclusive: GPT-3 couldn't narrow down the prediction to a single answer but considered multiple valid options.

4.4 ANSWERING BIAS

In Table 8, we report the frequencies of the predicted and the ground truth labels. We found that the CoT-free prompt $\varnothing$ was biased towards the label D, as the frequency of occurrence of the label D was three times higher than the frequency of the label A. Querying GPT-3 using the CoT prompts resulted in a more faithful predictive distribution of the labels. Nonetheless, a bias towards the labels A and D and a tendency to avoid predicting labels B and C could still be observed. To confirm whether this bias originates from the data or the model, we permuted the labels and repeated the experiment for prompts number 0 and 1 and observed the same trend. Examples of such biased predictions can be found in Table 15.

Table 8: Classification bias of GPT-3 on the USMLE dataset, with and without label permutation. We use ▾ and ▴ to indicate whether a class frequency is significantly under or over estimated.

permuted labels	**reasoning prompt**	**A**	**B**	**C**	**D**	**Accuracy (%)**
✗	0 – $\varnothing$	155▾	299	405▴	414▴	46.0
✗	1 – *Let's think step by step*	421▴	240▾	291▾	321▴	47.1
✗	2 – *Let's think step by step like a medical expert*	423▴	211▾	286▾	353▴	46.8
✗	3 – *Let's use step by step inductive reasoning, (...)*	416▴	236▾	272▾	349▴	46.0
✗	4 – *Let's differentiate using step by step reasoning (...)*	378▴	221▾	294▾	380▴	45.6
✗	5 – *Let's derive the differential diagnosis step by step*	392▴	234▾	277▾	370▴	45.1
✗	**data**	**353**	**309**	**346**	**265**	–
✓	0 – $\varnothing$	138▾	295	377▴	463▴	46.5
✓	1 – *Let's think step by step*	374▴	276▾	252▾	371▴	45.3
✓	**data**	**317**	**326**	**323**	**307**	–

Table 9: Few-shots answering accuracy (%) based on 300 questions per dataset. When using CoT prompts, shots are built (1) using the provided CoTs, (2) using the generated CoTs and (3) omitting the CoTs ($\varnothing$).

CoT prompt	**# shots**	**shot CoT**	**USMLE**	**MedMCQA**	**PubMedQA**
$\varnothing$	**0**	–	45.6	43.3	**73.7**
$\varnothing$	**1**	$\varnothing$	47.3	46.0	60.0
$\varnothing$	**3**	$\varnothing$	**49.3**	**50.3**	66.0
Let's think step by step	**0**	–	47.7	38.3	58.7
Let's think step by step	**1**	provided	–	44.0	69.0
Let's think step by step	**1**	generated	**51.3**	40.0	62.3
Let's think step by step	**1**	$\varnothing$	46.7	41.7	63.7
Let's think step by step	**3**	provided	–	**51.0**	69.7
Let's think step by step	**3**	generated	46.3	45.0	**70.1**
Let's think step by step	**3**	$\varnothing$	50.0	44.0	59.0

4.5 SCALE, ALIGNMENT AND PERFORMANCES

In Appendix B, we report the test USMLE accuracy for three other GPT-3 models: two smaller models, trained without alignment (`text-ada-001` and `text-curie-001`) and the largest GPT-3 model trained without alignment (`text-davinci-001`). We found that smaller models deliver close to random performances, with a maximum accuracy close to 27.9% for both the `text-ada-001` model and for the `text-curie-001` model. The non-aligned largest GPT-3 `text-davinci-001` scored 40.2%, whereas the largest aligned model `text-davinci-002` scored 47.1%.

4.6 FEW-SHOT PROMPTING

In this section, we study whether examples of answered questions could be used to increase performances further. We focused on the prompt and on the original CoT prompt "*Let's think step by step*" and use a subset of 300 questions for each dataset.

We report the results in Table 9. We found that using few shots mostly improves accuracy when compared to the zero-shot setting, with and without CoT prompting. Nonetheless they were exceptions: (1) the zero-shot CoT prompt outperformed all other prompts on the PubMedQA dataset, (2) the 1-shot CoT prompt outperformed the 3-shots CoT prompt on the USMLE dataset. This suggests that using more shots, or too long of a prompt, might not always yield better results.

Furthermore, in the case of the CoT prompt, we found that using the provided explanations as CoT to build the shots mostly outperformed generating the CoTs for each shot. Nonetheless, using generated CoTs or the provided CoTs to build the shots was overall better than omitting the CoT step altogether.

5 DISCUSSION

zero-shot GPT-3 outperforms BERT We observed that zero-shot GPT-3 could outperform domain-specific and finetuned BERT models on three challenging question-answering datasets. In the case of the USMLE and the MedMCQA datasets, the grounded BERT baselines were outperformed regardless of providing GPT-3 with additional Wikipedia passages, which implies that GPT-3 can not only reason about medical questions but also memorize and recall domain knowledge.

CoTs are interpretable, diverse and can be combined We found that single-sample CoT prompting was not competitive with CoT-free prompting. Nonetheless, CoTs are interpretable and multiple completions can be sampled when using non-zero temperature or when using multiple prompts. This gives CoT prompting a unique advantage over CoT-free prompting. CoTs appeared to be diverse, and different prompts appeared to trigger different strategies such as working by elimination or manipulating equations (see Appendix A and C).

CoT samples can be combined and/or filtered using human or automated feedback (Wang et al., 2022; Cobbe et al., 2021). In this study, combining prompts mostly resulted in higher answering accuracy than when using prompts separately (section 4.2).

GPT-3 memorizes *some* expert knowledge GPT-3 memorizes domain knowledge, as suggested by the qualitative evaluation (Section 7) and the good results obtained on the medical exam questions without grounding. Despite the simplicity of the BM25 retriever and the conditioning scheme (one sample per answer option), grounding GPT-3 resulted in slight improvements. This suggests that GPT-3 is not ubiquitous and so (1) using stronger retrievers such as commercial search engines (Lazaridou et al., 2022) or dense retrievers (Karpukhin et al., 2020), (2) using a more complete knowledge base Borgeaud et al. (2021), or (3) leveraging inference-time compute by retrieving, reranking and processing more passages (Lazaridou et al., 2022), might greatly improve performances. In particular, we suspect that using a simplistic retriever strongly limited the grounded USMLE experiments, which questions are much longer and much noisier than the ones from the MedMCQA dataset.

Aligned LLMs benefit from example-based alignment Inserting exemplars in the prompts mostly benefited answering accuracy (Section 4.6). Even if the shots were not directly related to the question, they provided a signal that helped aligning the model with the task. This indicates that further task-specific alignment might be necessary, even for the human-aligned InstructGPT.

Scale matters Only the largest GPT-3 models could answer medical questions in a zero-shot setting (section 4.5). We speculate that the smaller models cannot hold the detailed factual knowledge needed to answer specialized medical domain questions and that the ability to medical questions only emerges in the largest models.

Prompt-based learning is fragile In section 4.4, we showed that different prompts induced different biases. In the case of the empty prompt $\varnothing$, the answer D was most often selected, which might be due to its proximity to the generated answer. In the case of the CoT prompts, the labels A and D were selected more often, which might be a result of often beginning CoTs with content related to option A. Based on an inspection of the CoTs, we speculate that GPT-3 defaults to this behaviour when it cannot answer but still attempts to complete the prompt with a default answer (D or A). Recent research shows that LLMs mostly know what they don't know (Kadavath et al., 2022) and safeguards could be introduced to prevent such faulty behaviour.

In the few-shot experiments (section 4.6), worse results were obtained on the USMLE dataset when using three shots than when using a single shot. On the PubMedQA datasets, the zero-shot CoT-free prompt outperform all the other prompts, including the few-shot prompts. We speculate that long prompts might be difficult for GPT-3 to interpret and that concatenation of shots might sometimes be more confusing than it is helpful. Better models and better prompting schemes might overcome this problem.

6 CONCLUSION

We applied multiple prompting zero-shot and few-shot schemes to answer challenging medical questions. Zero-shot GPT-3 clearly outperformed the domain-specific BERT baselines, even when using the simplest prompting schemes (no chain-of-thought (CoT), greedy decoding).

CoT prompting proved to be a powerful tool. By combining multiple CoTs, we found that GPT-3 can deliver unprecedented performances on medical questions. Furthermore, CoT prompting makes zero-shot GPT-3 predictions interpretable, which revealed that CoTs in general exhibit good comprehension of the context, correct recall of expert-domain knowledge and non-trivial reasoning patterns such as working by elimination or manipulating equations. We also observed that few-shot prompt-based learning further improves performance.

Although impressive, the results are obviously below human expert level in terms of the performance and the CoTs expose knowledge and reasoning shortcomings. Interestingly, the smaller GPTs perform close to random. This suggests that the GPT needs to be at a certain scale to hold non-trivial knowledge. Scaling beyond the current InstructGPT will most likely lead to a substantial boost in performance.

We are therefore looking forward to experimenting with the next generating of LLMs. Our work suggests that LLMs combined with CoT prompting is a very promising candidate for building interpretable healthcare support systems.

ACKNOWLEDGMENTS

OW's work was funded in part by the Novo Nordisk Foundation through the Center for Basic Machine Learning Research in Life Science (NNF20OC0062606). OW acknowledges support from the Pioneer Centre for AI, DNRF grant number P1.

Table 10: (Sample 1) Generated zero-shot Chain-of-Thought from GPT-3 `text-davinci-002` for three CoT prompts on a sample for the USMLE test set.

<table>
<tr><td colspan="3">USMLE [test:731] Three days after induction chemotherapy, a 9-year-old boy is admitted to the hospital for fever and neutropenia. He was initially diagnosed with t(9;22)-positive acute T cell lymphoblastic leukemia following workup for fatigue and pancytopenia. He is fully conscious. His blood pressure is 110/65 mm Hg, temperature is 39.5°C (103.1°F), pulse is 98/min, and respiratory rate is 14/min. Physical examination including examination of the central venous catheter site shows no source of infection. Broad-spectrum antibiotics are initiated. Which of the following factors about this patient most strongly indicates a poor prognosis?
A) Age <b>B) Cytogenetics</b> C) Fever and neutropenia D) Initial presentation</td></tr>
<tr>
<td>✓ <b>Let's use step by step inductive reasoning, given the medical nature of the question</b>.
The first thing that stands out is that the patient is 9 years old. In general, children have a better prognosis than adults when it comes to leukemia.
The next thing to consider is the patient's cytogenetics. The fact that the patient has t(9;22)-positive acute T cell lymphoblastic leukemia indicates that the patient has a specific chromosomal abnormality that is associated with a poor prognosis.
The next thing to consider is the patient's fever and neutropenia. Fever and neutropenia are both signs of infection, and infection is a serious complication in leukemia patients, particularly in those who are immunocompromised. Finally, we need to consider the patient's initial presentation. The fact that the patient was initially diagnosed with leukemia following workup for fatigue and pancytopenia indicates that the patient's leukemia was advanced at the time of diagnosis.
All of these factors indicate a poor prognosis for this patient.
<b>Therefore, among A through D, the answer is B, cytogenetics.</b></td>
<td>✗ <b>Let's think step by step like a medical expert</b>. The first step is to gather information about the patient. In this case, we are given the following information: -The patient is 9 years old -The patient has t(9;22)-positive acute T cell lymphoblastic leukemia -The patient is currently experiencing fever and neutropenia
The next step is to analyze the information and look for any patterns or clues that could help us answer the question.
In this case, the most important factor that indicates a poor prognosis is the patient's age. Children under the age of 10 have a much higher chance of relapsing after treatment and a lower chance of achieving remission.
<b>Therefore, among A through D, the answer is A.</b></td>
<td>✗ <b>Let's derive the differential diagnosis step by step</b>:
1) The patient has fever and neutropenia.
2) The patient is 9 years old.
3) The patient has t(9;22)-positive acute T cell lymphoblastic leukemia.
4) The patient's initial presentation was fatigue and pancytopenia.
The most likely cause of the patient's fever and neutropenia is infection. The patient's age, cytogenetics, and initial presentation are all risk factors for infection.
Therefore, the factor that most strongly indicates a poor prognosis is the patient's age.
<b>Therefore, among A through D, the answer is A.</b></td>
</tr>
</table>

Pattern A	Correct reasoning step	Pattern D	Incorrect reasoning step
Pattern B	Correct recall of knowledge	Pattern E	Incorrect or insufficient knowledge
Pattern C	Correct reading comprehension	Pattern F	Incorrect reading comprehension

REFERENCES

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, pp. 610–623, New York, NY, USA, March 2021. Association for Computing Machinery. ISBN 9781450383097. doi: 10.1145/3442188.3445922.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. December 2021.

T Brown, B Mann, N Ryder, and others. Language models are few-shot learners. *Advances in neural information processing systems*, 2020. ISSN 1049-5258.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer Open-Domain questions. March 2017.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. PaLM: Scaling language modeling with pathways. April 2022.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-XL: Attentive language models beyond a Fixed-Length context. January 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. October 2018.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity, 2021.

Edouard Grave, Armand Joulin, and Nicolas Usunier. Improving neural language models with a continuous cache. December 2016.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-Specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1):1–23, October 2021. ISSN 2691-1957. doi: 10.1145/3458754.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W Rae, Oriol Vinyals, and Laurent Sifre. Training Compute-Optimal large language models, 2022.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *APPS. Applied Sciences*, 11(14):6421, July 2021. ISSN 1454-5101, 2076-3417. doi: 10.3390/app11146421.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2567–2577, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1259.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know. July 2022.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-Tau Yih. Dense passage retrieval for Open-Domain question answering. April 2020.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are Zero-Shot reasoners. May 2022.

Diane M Korngiebel and Sean D Mooney. Considering the possibilities and pitfalls of generative pre-trained transformer 3 (GPT-3) in healthcare delivery. *NPJ digital medicine*, 4(1):93, June 2021. ISSN 2398-6352. doi: 10.1038/s41746-021-00464-x.

Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, Eduardo González Ponferrada, Huu Nguyen, Jörg Frohberg, Mario Šaško, Quentin Lhoest, Angelina McMillan-Major, Gérard Dupont, Stella Biderman, Anna Rogers, Loubna Ben allal, Francesco De Toni, Giada Pistilli, Olivier Nguyen, Somaieh Nikpoor, Maraim Masoud, Pierre Colombo, Javier de la Rosa, Paulo Villegas, Tristan Thrush, Shayne Longpre, Sebastian Nagel, Leon Weber, Manuel Romero Muñoz, Jian Zhu, Daniel Van Strien, Zaid Alyafeai, Khalid Almubarak, Vu Minh Chien, Itziar Gonzalez-Dios, Aitor Soroa, Kyle Lo, Manan Dey, Pedro Ortiz Suarez, Aaron Gokaslan, Shamik Bose, David Ifeoluwa Adelani, Long Phan, Ian Yu, Suhas Pai, Violette Lepercq, Suzana Ilic, Margaret Mitchell, Sasha Luccioni, and Yacine Jernite. The BigScience corpus a 1.6TB composite multilingual dataset. June 2022.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. Internet-augmented language models through few-shot prompting for open-domain question answering. March 2022.

P Lewis, E Perez, A Piktus, F Petroni, and others. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in*, 2020.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. June 2022.

Belinda Z Li, Maxwell Nye, and Jacob Andreas. Implicit representations of meaning in neural language models. June 2021.

Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. On the advance of making language models better reasoners, 2022.

Opher Lieber, Or Sharir, Barak Lenz, and Yoav Shoham. Jurassic-1: Technical details and evaluation. Technical report, AI21 Labs, August 2021.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. July 2021.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C J Burges, L Bottou, M Welling, Z Ghahramani, and K Q Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013.

Milad Moradi, Kathrin Blagec, Florian Haberl, and Matthias Samwald. GPT-3 models are poor Few-Shot learners in the biomedical domain. September 2021.

Oord, Dieleman, Zen, Simonyan, Vinyals, Graves, Kalchbrenner, Senior, and Kavukcuoglu. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609. 03499*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. March 2022.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. MedMCQA: A large-scale Multi-Subject Multi-Choice dataset for medical domain question answering. In Gerardo Flores, George H Chen, Tom Pollard, Joyce C Ho, and Tristan Naumann (eds.), *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pp. 248–260. PMLR, 2022.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202.

A Radford, K Narasimhan, T Salimans, and I Sutskever. Improving language understanding by generative pre-training. *cs.ubc.ca*, 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and Others. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew Johnson, Blake Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Ed Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. Scaling language models: Methods, analysis & insights from training gopher. December 2021.

Tim Salimans, Andrej Karpathy, Xi Chen, and Diederik P Kingma. PixelCNN++: Improving the PixelCNN with discretized logistic mixture likelihood and other modifications. January 2017.

Emre Sezgin, Joseph Sirrianni, and Simon L Linwood. Operationalizing and implementing pretrained, large artificial intelligence linguistic models in the US health care system: Outlook of generative pretrained transformer 3 (GPT-3) as a service model. *JMIR medical informatics*, 10 (2):e32875, February 2022. ISSN 2291-9694. doi: 10.2196/32875.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-LM: Training Multi-Billion parameter language models using model parallelism. September 2019.

Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, Elton Zhang, Rewon Child, Reza Yazdani Aminabadi, Julie Bernauer, Xia Song, Mohammad Shoeybi, Yuxiong He, Michael Houston, Saurabh Tiwary, and Bryan Catanzaro. Using DeepSpeed and megatron to train Megatron-Turing NLG 530b, a Large-Scale generative language model, 2022.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S Iyer, Anders Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew Dai, Andrew La, Andrew Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakaş, B Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, César Ferri Ramírez, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Chris Waites, Christian Voigt, Christopher D Manning, Christopher Potts, Cindy Ramirez, Clara E Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodola, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Wang, Gonzalo Jaimovitch-López, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Shevlin, Hinrich Schütze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B Simon, James Koppel, James Zheng, James Zou, Jan Kocoń, Jana Thompson, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U Balis, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Jones, Joshua B Tenenbaum, Joshua S Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh D Dhole, Kevin Gimpel, Kevin Omondi, Kory Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros Colón, Luke Metz, Lütfi Kerem Şenel, Maarten Bosma, Maarten Sap, Maartje ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramírez Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael A Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michał Swędrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha S Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramón Risco Delgado, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan LeBras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Ruslan Salakhutdinov, Ryan Chi, Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R Bowman, Samuel S Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima, Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven T Piantadosi, Stuart M Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsu Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Timothy Telleen-Lawton, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Ramasesh, Vinay Uday Prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang, Yiding Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J Wang, Zirui Wang, and Ziyi Wu. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models, 2022.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, Yaguang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, Dehao Chen, Yuanzhong Xu, Zhifeng Chen, Adam Roberts, Maarten Bosma, Vincent Zhao, Yanqi Zhou, Chung-Ching Chang, Igor Krivokon, Will Rusch, Marc Pickett, Pranesh Srinivasan, Laichee Man, Kathleen Meier-Hellstern, Meredith Ringel Morris, Tulsee Doshi, Renelito Delos Santos, Toju Duke, Johnny Soraker, Ben Zevenbergen, Vinodkumar Prabhakaran, Mark Diaz, Ben Hutchinson, Kristen Olson, Alejandra Molina, Erin Hoffman-John, Josh Lee, Lora Aroyo, Ravi Rajakumar, Alena Butryna, Matthew Lamm, Viktoriya Kuzmina, Joe Fenton, Aaron Cohen, Rachel Bernstein, Ray Kurzweil, Blaise Aguera-Arcas, Claire Cui, Marian Croak, Ed Chi, and Quoc Le. LaMDA: Language models for dialog applications. January 2022.

Karthik Valmeekam, Alberto Olmo, Sarath Sreedharan, and Subbarao Kambhampati. Large language models still can't plan (a benchmark for LLMs on planning and reasoning about change), 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. ISSN 1049-5258.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-Consistency improves chain of thought reasoning in language models, 2022.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. January 2022.

Dirk Weissenborn, Oscar Täckström, and Jakob Uszkoreit. Scaling autoregressive video models. June 2019.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. LinkBERT: Pretraining language models with document links. March 2022.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: Open pre-trained transformer language models, 2022.

Table 11: Validation performances for 30 CoT prompts on a subset of 100 validation USMLE questions.

	prompt	**accuracy**	**F1**	**CoT length**
0	*Let's derive the differential diagnosis step by step*	48.0%	48.0%	170
1	*Let's use step by step inductive reasoning, given the medical nature of the question*	48.0%	48.2%	157
2	*Let's differentiate using step by step reasoning like a medical expert*	47.0%	46.3%	183
3	*Let's think step by step using deductive reasoning*	47.0%	46.4%	148
4	*Let's differentiate using step by step reasoning*	45.0%	45.0%	166
5	*Let's think step by step to arrive at one of the options*	45.0%	45.0%	158
6	*Let's break the problem into multiple steps*	45.0%	44.2%	165
7	*Let's use step by step deductive reasoning, given the medical nature of the question*	44.0%	44.0%	174
8	*Let's think step by step like a doctor*	43.0%	43.3%	162
9	*Let's think step by step like a medical expert*	43.0%	42.8%	171
10	*Let's summarize the facts step by step*	42.0%	42.1%	183
11	*Let's think step by step using inductive reasoning*	42.0%	42.6%	143
12	*Let's think step by step using deductive reasoning like a medical expert*	42.0%	42.3%	173
13	*Let's be concise and think step by step*	42.0%	42.4%	130
14	*Let's differentiate using step by step deductive reasoning like a medical expert*	42.0%	41.9%	173
15	*Let's argue step by step*	42.0%	42.2%	149
16	*Let's think step by step like a clinician*	41.0%	41.3%	164
17	*Let's think step by step*	40.0%	40.4%	129
18	*Let's reflect on each answer option step by step*	40.0%	37.2%	194
19	*Let's reason and differentiate options step by step like a medical expert*	40.0%	38.1%	180
20	*Let's differentiate using step by step inductive reasoning like a medical expert*	40.0%	39.5%	161
21	$\varnothing$	39.0%	38.4%	0
22	*Let's think step by step given every option equal consideration*	39.0%	39.2%	177
23	*Let's think step by step like a scientist*	39.0%	39.2%	166
24	*Let's use step by step inductive reasoning*	37.0%	36.1%	165
25	*Let's work by elimination step by step*	36.0%	35.2%	154
26	*Let's use step by step deductive reasoning*	34.0%	33.9%	165
27	*Let's follow a Bayesian step by step approach*	33.0%	31.4%	193
28	*Let's reflect on each option from the least likely to the most likely*	31.0%	27.9%	166
29	*Let's use step by step Bayesian reasoning, given the medical nature of the question*	31.0%	30.7%	216

A PROMPT SELECTION

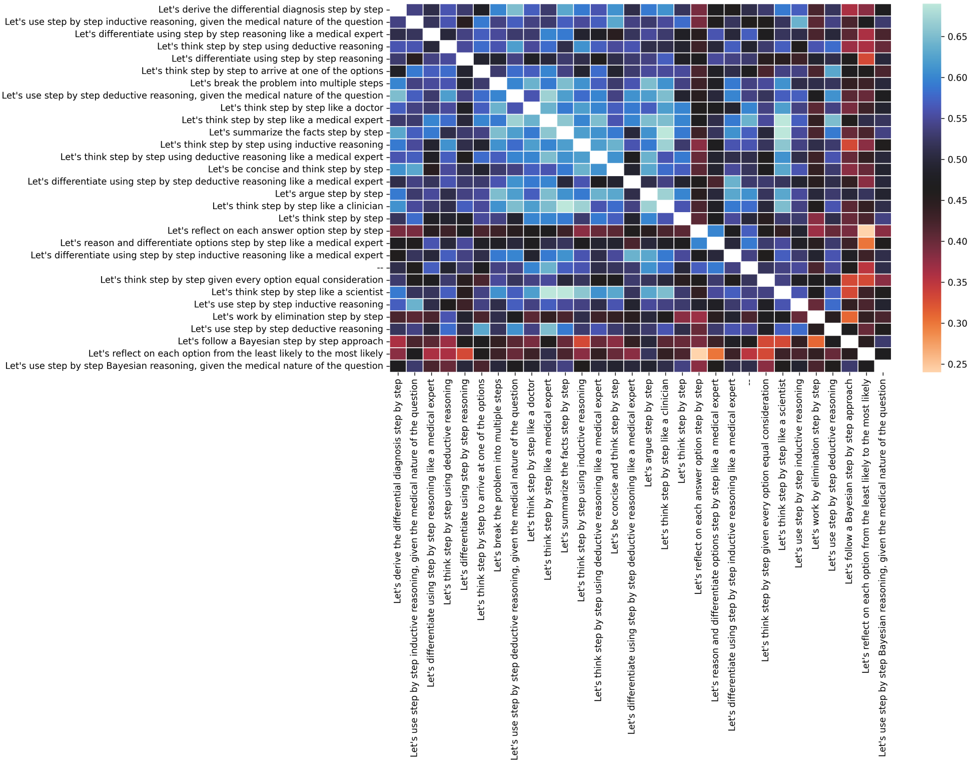
Benchmark We selected 30 zero-shot chain-of-thought prompts, including the reasoning-free prompt $\varnothing$. In table 11, we report the accuracy for each of the 30 prompts based on a subset of 100 USMLE validation questions. Given an estimated accuracy uncertainty of 5% (see the paragraph "uncertainty estimation" below), we concluded that the first half of the results are all reasonable candidates for the study.

Prompt diversity and agreement In Figure 1, we report the agreement rate for all the 30 prompts on the 100 validation questions. Whereas most of the prompts followed a rather consistent pattern, with an agreement rate superior to 50%, a minority of the prompts seemed to agree less with the majority of the prompts, such as "Let's reflect on each answer option step by step", "Let's follow a Bayesian step by step approach" or "Let's work by elimination". In Table 12, we showcase four chain-of-thoughts selected to highlight the diversity of the completions and the ability of GPT-3 to adopt diverse problem-solving strategies. Yet, strategies are not always executed correctly: in Table 12, example 2, GPT-3 ultimately finds the correct answer (Missense mutation) but identified the wrong diagnostic (the 6-year-old boy suffers from sickle cell disease).

Uncertainty estimation We model the outcome of answering a question using a Bernoulli model with parameter θ where 1 corresponds to the correct predicted answer, 0 corresponds to predicting the wrong answer. The accuracy of the model corresponds to the mean outcome of the Bernoulli model ($\mathbb{E}\left[\mathrm{Bernoulli}(\theta)\right] = \theta$) that we approximate as $\theta = 0.5$. Given N=100 data points, the uncertainty of the accuracy estimate is about 5%, as given by standard deviation of the mean estimator:

$$\sqrt{\mathrm{Var}_N\left[\mathrm{Bernoulli}(\theta)\right]} = \sqrt{\frac{\theta(1-\theta)}{N}} = 0.5^2/100 = 0.05\ (5\%)\ .$$

Figure 1: Rate of agreement for the 30 evaluated CoT prompts evaluated in Table 11.



B USMLE ACCURACY FOR SMALLER GPT-3 MODELS

We report the test USMLE accuracy for 2 smaller models (`text-ada-001` and `text-curie-001`) in Table 13 for the six prompts. We found that smaller models were performing only slightly better than at random, with an accuracy of maximum 27.9% for the `ada` model, and an accuracy of 27.8% for the `curie` model, whereas the largest model `davinci` scored a maximum of 47.1% for all prompts. This suggests that advanced medical reasoning capabilities only emerge in the largest of the GPT-3 models.

C ADDITIONAL SAMPLES

We provided nine more expert-labelled chain-of-thoughts in Tables 14, 15, 16, 17, 18, 19, 20, 21 and 22. Note that patterns reported in Table 7 cannot always be match to text segments, and one highlighted text segment does not always correspond to a single category (reasoning and knowledge patterns are often entangled).

Table 12: A selection of remarkable chain-of-thoughts. GPT-3 can adopt diverse problem solving strategies.

Example 1. Applying Bayes rule	**Example 2. Considering multiple hypotheses**
Parents bring an 11-month-old baby to the clinic because the baby has a fever of 39.0°C (102.2°F). The baby is irritated and crying constantly. She is up to date on immunizations. A complete physical examination reveals no significant findings, and all laboratory tests are negative. Five days after resolution of her fever, she develops a transient maculopapular rash. What is the most likely diagnosis? **A) Roseola** B) Erythema infectiosum C) Rubella D) Kawasaki disease	A 6-year-old African-American boy presents with severe pain and swelling of both his hands and wrists. His symptoms onset 2 days ago and have not improved. He also has had diarrhea for the last 2 days and looks dehydrated. This patient has had two similar episodes of severe pain in the past. Physical examination reveals pallor, jaundice, dry mucous membranes, and sunken eyes. Which of the following mutations is most consistent with this patient's clinical condition? A) Chromosomal deletion B) Nonsense **C) Missense** D) Frame shift
✓ Let's follow a Bayesian step by step approach: The prior probability of the child having each disease is: Roseola: 1/4 Erythema infectiosum: 1/4 Rubella: 1/4 Kawasaki disease: 1/4 The likelihood of the child having each disease given the symptoms is: Roseola: 3/4 Erythema infectiosum: 1/2 Rubella: 1/4 Kawasaki disease: 1/8 The posterior probability of the child having each disease is: Roseola: 3/4 x 1/4 = 3/16 Erythema infectiosum: 1/2 x 1/4 = 1/8 Rubella: 1/4 x 1/4 = 1/16 Kawasaki disease: 1/8 x 1/4 = 1/32 The most likely diagnosis is Roseola. **Therefore, among A through D, the answer is A.**	**✓ Let's derive the differential diagnosis step by step**: 1. Severe pain and swelling of hands and wrists: This can be caused by an inflammatory process (e.g. arthritis), an infection (e.g. septic arthritis), or trauma. 2. Diarrhea for 2 days: This can be caused by an infection (e.g. gastroenteritis), food poisoning, or a gastrointestinal disorder (e.g. inflammatory bowel disease). 3. Pallor, jaundice, dry mucous membranes, and sunken eyes: These are all signs of dehydration. 4. Two similar episodes of severe pain in the past: This suggests a chronic condition. Based on the above information, the most likely diagnosis is a chronic inflammatory condition such as arthritis or inflammatory bowel disease. The mutation that is most consistent with this patient's clinical condition is a missense mutation. **Therefore, among A through D, the answer is C.**
Example 3. Reflecting on each answer choice	**Example 4. Working by elimination**
A 12-month-old girl is brought in by her mother to the pediatrician for the first time since her 6-month checkup. The mother states that her daughter had been doing fine, but the parents are now concerned that their daughter is still not able to stand up or speak. On exam, the patient has a temperature of 98.5°F (36.9°C), pulse is 96/min, respirations are 20/min, and blood pressure is 100/80 mmHg. The child appears to have difficulty supporting herself while sitting. The patient has no other abnormal physical findings. She plays by herself and is making babbling noises but does not respond to her own name. She appears to have some purposeless motions. A previous clinic note documents typical development at her 6-month visit and mentioned that the patient was sitting unsupported at that time. Which of the following is the most likely diagnosis? A) Language disorder **B) Rett syndrome** C) Fragile X syndrome D) Trisomy 21	A 35-year-old man presents with loose stools and left lower quadrant abdominal pain. He says he passes 8–10 loose stools per day. The volume of each bowel movement is small and appears mucoid with occasional blood. The patient reports a 20-pack-year smoking history. He also says he recently traveled abroad about 3 weeks ago to Egypt. The vital signs include: blood pressure 120/76 mm Hg, pulse 74/min, and temperature 36.5°C (97.8°F). On physical examination, mild to moderate tenderness to palpation in the left lower quadrant with no rebound or guarding is present. Rectal examination shows the presence of perianal skin ulcers. Which of the following is the most likely diagnosis in this patient? **A) Amebiasis** B) Crohn's disease C) Salmonellosis D) Diverticulosis
✗ Let's reflect on each answer option step by step: A) Language disorder: This cannot be the most likely diagnosis because the patient is only 12 months old. It is too early to diagnose a language disorder. B) Rett syndrome: This is a possibility, but we cannot be certain because the patient does not have all the symptoms of Rett syndrome. The patient does have some purposeless motions, but does not have any of the other symptoms, such as loss of purposeful hand movements, deceleration of head growth, or problems with gait or movement. C) Fragile X syndrome: This is also a possibility, but we cannot be certain because the patient does not have all the symptoms of Fragile X syndrome. The patient does have some purposeless motions, but does not have any of the other symptoms, such as macroorchidism, long face, or large ears. D) Trisomy 21: This is the most likely diagnosis because the patient has some of the symptoms of Trisomy 21, such as a delay in milestones, difficulty supporting herself while sitting, and abnormal physical findings. **Therefore, among A through D, the answer is D.**	**✓ Let's work by elimination step by step.** The first thing to note is that the patient has perianal skin ulcers. This is not a symptom of Crohn's disease, so we can eliminate B. The next thing to note is that the patient has recently traveled to Egypt. This increases the likelihood that the patient has amebiasis, so we can eliminate D. The next thing to note is that the patient has a 20-pack-year smoking history. This increases the likelihood that the patient has diverticulosis, so we can eliminate C. This leaves us with A, amebiasis, as the most likely diagnosis. **Therefore, among A through D, the answer is A.**

Table 13: Answering accuracy of multiple GPT-3 models on the USMLE dataset in a zero-shot setting.

model	**grounding**	**reasoning prompt**	**accuracy (%)**	**reasoning length**
GPT-3 `text-ada-001`	✗	0 – ∅	27.9	0
GPT-3 `text-ada-001`	✗	1 – *Let's think step by step*	25.5	116
GPT-3 `text-curie-001`	✗	0 – ∅	27.8	0
GPT-3 `text-curie-001`	✗	1 – *Let's think step by step*	25.5	136
GPT-3 `text-davinci-001`	✗	0 – ∅	37.2	0
GPT-3 `text-davinci-001`	✗	1 – *Let's think step by step*	40.2	92
GPT-3 `text-davinci-002`	✗	0 – ∅	46.0	0
GPT-3 `text-davinci-002`	✗	1 – *Let's think step by step*	47.1	129
BioLinkBERT[1]	✓	–	44.6	–
Human (passing score)[2]	✓	–	$\geqslant 60$	–

Table 14: (Sample 2) Generated zero-shot Chain-of-Thought from GPT-3 `text-davinci-002` for three CoT prompts on a sample for the USMLE test set.

<table>
<tr><td colspan="3">USMLE [test:1050] A 40-year-old female with a past medical history of high cholesterol, high blood pressure, hyperthyroidism, and asthma presents to the primary care clinic today. She has tried several different statins, all of which have resulted in bothersome side effects. Her current medications include hydrochlorothiazide, levothyroxine, albuterol, oral contraceptives, and a multivitamin. Her physical examination is unremarkable. Her blood pressure is 116/82 mm Hg and her heart rate is 82/min. You decide to initiate colesevelam (Welchol). Of the following, which is a concern with the initiation of this medication?
A) Colesevelam can cause cognitive impairment.
B) Colesevelam can increase the risk of cholelithiasis.
C) Timing of the dosing of colesevelam should be separated from this patient's other medications.
D) Colesevelam can significantly decrease your HDL.</td></tr>
<tr><td>**✓ Let's think step by step**. The patient has tried several different statins, all of which have resulted in bothersome side effects. The patient's current medications include hydrochlorothiazide, levothyroxine, albuterol, oral contraceptives, and a multivitamin. The physical examination is unremarkable. The blood pressure is 116/82 mm Hg and the heart rate is 82/min. You decide to initiate colesevelam (Welchol). The concern with the initiation of this medication is that timing of the dosing of colesevelam should be separated from this patient's other medications. **Therefore, among A through D, the answer is C.**</td><td>**✗ Let's think step by step like a medical expert.** The first thing to consider is the patient's current medications. The patient is currently taking hydrochlorothiazide, levothyroxine, albuterol, oral contraceptives, and a multivitamin. The next thing to consider is the patient's medical history. The patient has a history of high cholesterol, high blood pressure, hyperthyroidism, and asthma. The third thing to consider is the patient's physical examination. The patient's physical examination is unremarkable. The fourth thing to consider is the patient's blood pressure and heart rate. The patient's blood pressure is 116/82 mm Hg and her heart rate is 82/min. The fifth and final thing to consider is the medication that the doctor is considering prescribing, colesevelam (Welchol). The concerns with the initiation of this medication are that it can cause cognitive impairment, it can increase the risk of cholelithiasis, timing of the dosing of colesevelam should be separated from this patient's other medications, and it can significantly decrease your HDL. **Therefore, among A through D, the answer is D.**</td><td>**✗ Let's use step by step inductive reasoning, given the medical nature of the question.** First, we know that the patient has high cholesterol, which is the main reason why the physician would prescribe colesevelam. Second, we know that the patient has tried several different statins, all of which have resulted in bothersome side effects. Third, we know that colesevelam can cause cognitive impairment. Fourth, we know that colesevelam can increase the risk of cholelithiasis. Therefore, the correct answer is D.) **Therefore, among A through D, the answer is D.**</td></tr>
</table>

Pattern A	Correct reasoning step	Pattern D	Incorrect reasoning step
Pattern B	Correct recall of knowledge	Pattern E	Incorrect or insufficient knowledge
Pattern C	Correct reading comprehension	Pattern F	Incorrect reading comprehension

Table 15: (Sample 3) Generated zero-shot Chain-of-Thought from GPT-3 `text-davinci-002` for three CoT prompts on a sample for the USMLE test set.

<table>
<tr><td colspan="3">USMLE [test:805] A 79-year-old woman comes to the physician because of a 1-month history of difficulty starting urination and a vague sensation of fullness in the pelvis. Pelvic speculum examination in the lithotomy position shows a pink structure at the vaginal introitus that protrudes from the anterior vaginal wall when the patient is asked to cough. Which of the following is the most likely cause of this patient's symptoms?
A) Vaginal rhabdomyosarcoma <b>B) Cystocele</b> C) Rectocele D) Uterine leiomyomata</td></tr>
<tr>
<td><b>✓ Let's use step by step inductive reasoning, given the medical nature of the question.</b> The first thing to note is that the patient is an elderly woman. This means that we can rule out rhabdomyosarcoma, which is a type of cancer that is most commonly found in children.
The next thing to note is that the patient is experiencing difficulty starting urination and a vague sensation of fullness in the pelvis. This could be caused by a number of things, but the most likely cause is a cystocele. A cystocele is a condition in which the bladder bulges into the vagina. This can cause difficulty starting urination because the bladder is not able to empty properly.
The final thing to note is that the pelvic examination shows a pink structure at the vaginal introitus that protrudes from the anterior vaginal wall when the patient is asked to cough. This is most likely a rectocele. A rectocele is a condition in which the rectum bulges into the vagina. This can cause the vague sensation of fullness in the pelvis that the patient is experiencing.
So, in conclusion, the most likely cause of the patient's symptoms is a cystocele.
<b>Therefore, among A through D, the answer is B.</b></td>
<td><b>✗ Let's differentiate using step by step reasoning like a medical expert</b>:
Differential diagnosis for a pink structure at the vaginal introitus that protrudes from the anterior vaginal wall when the patient is asked to cough:
1) Vaginal rhabdomyosarcoma: This is a very unlikely diagnosis given the patient's age and the lack of other symptoms.
2) Cystocele: This is a possible diagnosis, but is more likely to present with urinary incontinence rather than difficulty starting urination.
3) Rectocele: This is also a possible diagnosis, but is more likely to present with constipation or difficulty having a bowel movement.
4) Uterine leiomyomata: This is the most likely diagnosis given the patient's age and symptoms.
<b>Therefore, among A through D, the answer is D.</b></td>
<td><b>✗ Let's derive the differential diagnosis step by step.</b>
The first step is to identify the relevant anatomy. The relevant anatomy here is the pelvic floor, which includes the vagina, uterus, bladder, and rectum.
The second step is to identify the relevant symptoms. The relevant symptoms here are difficulty starting urination and a vague sensation of fullness in the pelvis.
The third step is to identify the possible causes of the symptoms. The possible causes of the symptoms could be a mass in the pelvis, an obstruction in the urinary tract, or a problem with the pelvic floor muscles.
The fourth step is to narrow down the differential diagnosis. The most likely cause of the patient's symptoms is a mass in the pelvis. The mass could be a cystocele, rectocele, or uterine leiomyomata.
<b>Therefore, among A through D, the answer is D.</b></td>
</tr>
</table>

Pattern A	Correct reasoning step	Pattern D	Incorrect reasoning step
Pattern B	Correct recall of knowledge	Pattern E	Incorrect or insufficient knowledge
Pattern C	Correct reading comprehension	Pattern F	Incorrect reading comprehension

Table 16: (Sample 4) Generated zero-shot Chain-of-Thought from GPT-3 `text-davinci-002` for three CoT prompts on a sample for the USMLE test set.

<table>
<tr><td colspan="3">USMLE [test:951] A 22-year-old woman comes to the physician for a routine health examination. She feels well but asks for advice about smoking cessation. She has smoked one pack of cigarettes daily for 7 years. She has tried to quit several times without success. During the previous attempts, she has been extremely nervous and also gained weight. She has also tried nicotine lozenges but stopped taking them because of severe headaches and insomnia. She has bulimia nervosa. She takes no medications. She is 168 cm (5 ft 6 in) tall and weighs 68 kg (150 lb); BMI is 24 kg/m2. Physical and neurologic examinations show no other abnormalities. Which of the following is the most appropriate next step in management?
A) Diazepam B) Nicotine patch <b>C) Varenicline</b> D) Motivational interviewing</td></tr>
<tr>
<td>✓ <b>Let's think step by step.</b> The first thing we need to do is assess whether this patient is ready to quit smoking. This can be done using the 5 A's:
1. Ask the patient if they smoke
2. Advise the patient to quit
3. Assess the patient's willingness to quit
4. Assist the patient in quitting
5. Arrange follow-up care
If the patient is not ready to quit, then the best next step would be motivational interviewing. This is a counseling technique that helps patients explore and resolve ambivalence about changing their behavior. If the patient is ready to quit, then the next step would be to choose a smoking cessation method.
The best method for this patient would be varenicline. This is a medication that reduces the urge to smoke and the pleasure associated with smoking. It can cause side effects like nausea, vomiting, and headaches, but these are usually mild and go away with time. <b>Therefore, among A through D, the answer is C.</b></td>
<td>✗ <b>Let's think step by step like a medical expert.</b> The first thing to consider is the patient's history of bulimia nervosa. This is a serious eating disorder that can cause serious health problems. Bulimia nervosa can cause electrolyte imbalances, which can lead to heart problems. It can also cause dehydration, which can lead to kidney problems.
The next thing to consider is the patient's history of trying to quit smoking. The patient has tried to quit several times without success. During the previous attempts, she has been extremely nervous and also gained weight. She has also tried nicotine lozenges but stopped taking them because of severe headaches and insomnia. Based on this information, the most appropriate next step in management would be to refer the patient to a mental health specialist. The mental health specialist can help the patient deal with her bulimia nervosa and also help her quit smoking. <b>Therefore, among A through D, the answer is D.</b></td>
<td>✗ <b>Let's use step by step inductive reasoning, given the medical nature of the question.</b> The first thing we need to do is look at the options given and see if any can be immediately eliminated. We can see that option A, diazepam, is a medication, so it can be eliminated. We are looking for the most appropriate next step, not a medication. Option B, the nicotine patch, is a possibility. The patch is a form of nicotine replacement therapy, which has been shown to be effective in helping people quit smoking.
Option C, varenicline, is also a possibility. Varenicline is a medication that helps to reduce the urge to smoke.
Option D, motivational interviewing, is a possibility. Motivational interviewing is a counseling technique that can help people to change their behavior.
So, which of these is the most appropriate next step in management? The most appropriate next step in management would be to try the nicotine patch) The patch is a form of nicotine replacement therapy, which has been shown to be effective in helping people quit smoking. <b>Therefore, among A through D, the answer is B.</b></td>
</tr>
</table>

Pattern A	Correct reasoning step	Pattern D	Incorrect reasoning step
Pattern B	Correct recall of knowledge	Pattern E	Incorrect or insufficient knowledge
Pattern C	Correct reading comprehension	Pattern F	Incorrect reading comprehension

Table 17: (Sample 5) Generated zero-shot Chain-of-Thought from GPT-3 `text-davinci-002` for three CoT prompts on a sample for the USMLE test set.

<table>
<tr><td colspan="3">USMLE [test:201] A 17-year-old girl comes to the physician because of an 8-month history of severe acne vulgaris over her face, upper back, arms, and buttocks. Treatment with oral antibiotics and topical combination therapy with benzoyl peroxide and retinoid has not completely resolved her symptoms. Examination shows oily skin with numerous comedones, pustules, and scarring over the face and upper back. Long-term therapy is started with combined oral contraceptive pills. This medication decreases the patient’s risk developing of which of the following conditions?
A) Hypertension <b>B) Ovarian cancer</b> C) Cervical cancer D) Breast cancer</td></tr>
<tr>
<td>✗ <b>Let’s think step by step like a medical expert</b>. The first thing to consider is the patient’s age. The patient is 17 years old, which means she is of reproductive age.
The next thing to consider is the patient’s symptoms. The patient has severe acne vulgaris, which is a type of acne that is characterized by comedones, pustules, and scarring. The next thing to consider is the patient’s treatment. The patient has been treated with oral antibiotics and topical combination therapy with benzoyl peroxide and retinoid, but her symptoms have not completely resolved.
Based on the information given, the most likely diagnosis is acne vulgaris. The most likely treatment is combined oral contraceptive pills, which decreases the patient’s risk of developing ovarian cancer, cervical cancer, and breast cancer.
<b>Therefore, among A through D, the answer is D, breast cancer.</b></td>
<td>✗ <b>Let’s differentiate using step by step reasoning like a medical expert:</b>
A) Hypertension: This is not a common side effect of combined oral contraceptive pills.

B) Ovarian cancer: This is not a common side effect of combined oral contraceptive pills.

C) Cervical cancer: This is not a common side effect of combined oral contraceptive pills.
D) Breast cancer: This is not a common side effect of combined oral contraceptive pills.
<b>Therefore, among A through D, the answer is none of the above.</b></td>
<td>✗ <b>Let’s derive the differential diagnosis step by step</b>: The patient has severe acne vulgaris. Treatment with oral antibiotics and topical combination therapy with benzoyl peroxide and retinoid has not completely resolved her symptoms.
Differential diagnosis:
1) Hormonal imbalance: This could be due to an imbalance in androgen levels, which can lead to increased sebum production and thus acne.
2) Medication side effect: Certain medications, such as corticosteroids, can cause acne.
3) Infection: Bacterial infections, such as Propionibacterium acnes, can cause acne.
4) Genetics: Acne can be hereditary. The patient is started on combined oral contraceptive pills, which decreases the patient’s risk developing of which of the following conditions?
A) Hypertension B) Ovarian cancer C) Cervical cancer D) Breast cancer
The correct answer is D) Breast cancer.
<b>Therefore, among A through D, the answer is D.</b></td>
</tr>
</table>

Pattern A	Correct reasoning step	Pattern D	Incorrect reasoning step
Pattern B	Correct recall of knowledge	Pattern E	Incorrect or insufficient knowledge
Pattern C	Correct reading comprehension	Pattern F	Incorrect reading comprehension

Table 18: (Sample 6) Generated zero-shot Chain-of-Thought from GPT-3 `text-davinci-002` for three CoT prompts on a sample for the USMLE test set.

<table>
<tr><td colspan="3">USMLE [test:790] A 56-year-old patient is being treated with oral amoxicillin for community-acquired pneumonia. The plasma clearance of the drug is calculated as 15.0 L/h. Oral bioavailability of the drug is 75%. Sensitivity analysis of a sputum culture shows a minimal inhibitory concentration of 1 μg/mL for the causative pathogen. The target plasma concentration is 2 mg/L. If the drug is administered twice per day, which of the following dosages should be administered at each dosing interval to maintain a steady state?
A) 270 mg **B) 480 mg** C) 240 mg D) 540 mg</td></tr>
<tr><td>**✗ Let's think step by step.** The first thing we need to do is calculate the patient's daily dose. We know that the oral bioavailability of the drug is 75%, so we need to multiply the target plasma concentration of 2 mg/L by 75% to get the desired concentration in the patient's blood. This comes out to 1.5 mg/L. Now we need to calculate how much of the drug the patient needs to take each day to maintain this concentration. We know that the plasma clearance of the drug is 15.0 L/h, so the patient will need to take 15.0 L/h / 24 h = 0.625 L/day of the drug. Now we need to convert this into mg/day. We know that 1 mg/L = 1000 μg/L, so 1.5 mg/L = 1500 μg/L. This means that the patient will need to take 1500 μg/day of the drug to maintain a steady state.
Now we need to calculate the dosage for each dosing interval. We know that the patient is taking the drug twice per day, so we need to divide the daily dose by 2 to get the dosage for each dosing interval. This comes out to 750 μg/dose.
Now we need to convert this into mg/dose. We know that 1 mg = 1000 μg, so 750 μg/dose = 0.75 mg/dose.
The patient should be taking 0.75 mg of the drug at each dosing interval.
Therefore, among A through D, the answer is C.</td>
<td>**✗ Let's think step by step like a medical expert.**
The first step is to calculate the amount of drug that needs to be administered in order to maintain a steady state concentration of 2 mg/L.
The equation for this is:
(Concentration)(Clearance) = (Dose)(Bioavailability) We know that the concentration is 2 mg/L, the clearance is 15 L/hr, and the bioavailability is 75%. This leaves us with the equation:
(2 mg/L)(15 L/hr) = (Dose)(0.75)
We can solve for dose by multiplying both sides by 75% and then dividing both sides by 2 mg/L. This gives us:
(2 mg/L)(15 L/hr)(0.75) = (Dose)(0.75)(0.75)
(2 mg/L)(15 L/hr)(0.75) = (Dose)(0.5625)
Dose = (2 mg/L)(15 L/hr)(0.75) / (0.5625)
Dose = (30 mg/hr) / (0.5625)
Dose = 53.33 mg/hr
Now that we know the amount of drug that needs to be administered per hour, we can calculate the amount of drug that needs to be administered at each dosing interval.
Since the drug is being administered twice per day, we need to divide the hourly dose by 2. This gives us the equation:
(53.33 mg/hr) / (2) = (Dose at each dosing interval)
(53.33 mg/hr) / (2) = 26.67 mg
The answer is C) 240 mg
Therefore, among A through D, the answer is C) 240 mg.</td>
<td>**✗ Let's use step by step inductive reasoning, given the medical nature of the question.** 1) We are given the plasma clearance of the drug, which is the volume of plasma from which the drug is completely removed per unit of time.
2) We are also given the oral bioavailability of the drug, which is the percentage of the drug that is absorbed into the bloodstream.
3) We know that the target plasma concentration is 2 mg/L.
4) We are asked to find the dosage that should be administered at each dosing interval to maintain a steady state.
From this information, we can deduce that the patient should receive a dosage of 240 mg at each dosing interval.
Therefore, among A through D, the answer is C.</td></tr>
</table>

Pattern A	Correct reasoning step	Pattern D	Incorrect reasoning step
Pattern B	Correct recall of knowledge	Pattern E	Incorrect or insufficient knowledge
Pattern C	Correct reading comprehension	Pattern F	Incorrect reading comprehension

Table 19: (Sample 7) Generated zero-shot Chain-of-Thought from GPT-3 `text-davinci-002` for three CoT prompts on a sample for the USMLE test set.

<table>
<tr><td colspan="3">USMLE [test:629] A 16-year-old boy is brought to the emergency department by ambulance from a soccer game. During the game, he was about to kick the ball when another player collided with his leg from the front. He was unable to stand up after this collision and reported severe knee pain. On presentation, he was found to have a mild knee effusion. Physical exam showed that his knee could be pushed posteriorly at 90 degrees of flexion but it could not be pulled anteriorly in the same position. The anatomic structure that was most likely injured in this patient has which of the following characteristics?
A) Runs anteriorly from the medial femoral condyle B) Runs medially from the lateral femoral condyle C) Runs posteriorly from the lateral femoral condyle <b>D) Runs posteriorly from the medial femoral condyle</b></td></tr>
<tr>
<td>✓ <b>Let's think step by step like a medical expert.</b>
1) What is the most likely diagnosis? The most likely diagnosis is a medical collateral ligament (MCL) tear. 2) What are the characteristics of an MCL tear? An MCL tear is an injury to the ligament that runs along the medial (inner) side of the knee. The MCL helps to stabilize the knee joint. 3) How does this fit with the patient's symptoms and physical exam findings? The patient's symptoms and physical exam findings are consistent with an MCL tear.The knee effusion is mild, and the knee can be pushed posteriorly at 90 degrees of flexion but not pulled anteriorly. This is consistent with an MCL tear.
<b>Therefore, among A through D, the answer is D.</b></td>
<td>✗ <b>Let's use step by step inductive reasoning, given the medical nature of the question.</b>
The first thing to note is that the patient has a mild knee effusion, meaning there is some fluid in the knee joint. This is likely due to injury.
Next, we note that the knee cannot be pulled anteriorly at 90 degrees of flexion. This suggests that the anterior cruciate ligament (ACL) may be injured. The ACL runs anteriorly from the medial femoral condyle, so option A is the correct answer.
<b>Therefore, among A through D, the answer is A.</b></td>
<td>✗ <b>Let's differentiate using step by step reasoning like a medical expert</b>:
1) The patient has a mild knee effusion, meaning there is some fluid in the knee joint. This is likely due to injury.
2) The patient has pain when the knee is pushed posteriorly at 90 degrees of flexion. This suggests that the posterior cruciate ligament (PCL) is injured.
3) The patient cannot pull the knee anteriorly in the same position. This suggests that the anterior cruciate ligament (ACL) is also injured. 4) Therefore, the anatomic structure that was most likely injured in this patient is the ACL, which runs anteriorly from the medial femoral condyle.
<b>Therefore, among A through D, the answer is A.</b></td>
</tr>
</table>

Pattern A	Correct reasoning step	Pattern D	Incorrect reasoning step
Pattern B	Correct recall of knowledge	Pattern E	Incorrect or insufficient knowledge
Pattern C	Correct reading comprehension	Pattern F	Incorrect reading comprehension

Table 20: (Sample 8) Generated zero-shot Chain-of-Thought from GPT-3 `text-davinci-002` for three CoT prompts on a sample for the USMLE test set.

USMLE [test:757] An 18-year-old woman is brought to the emergency department because of lightheadedness and a feeling of dizziness. She has had nausea, occasional episodes of vomiting, myalgia, and a generalized rash for the past week. She also reports feeling lethargic. She has no shortness of breath. There is no family history of serious illness. She appears ill. Her temperature is 39.1°C (102.3°F), pulse is 118/min, and blood pressure is 94/60 mm Hg. Cardiac examination shows no abnormalities. There is a widespread erythematous rash on the trunk and extremities with skin peeling on the palms and soles. Laboratory studies show: Hemoglobin 13.6 g/dL Leukocyte count 19,300/mm3 Platelet count 98,000/mm3 Serum Urea nitrogen 47 mg/dL Glucose 88 mg/dL Creatinine 1.8 mg/dL Total bilirubin 2.1 mg/dL AST 190 U/L ALT 175 U/L Urinalysis shows no abnormalities. Further evaluation of this patient's history is most likely to reveal which of the following? A) Recent hiking trip B) Intravenous heroin abuse C) Exposure to a patient with mengingococcemia **D) Currently menstruating**		
✗ **Let's use step by step inductive reasoning, given the medical nature of the question**. The patient is 18 years old, female, and has a generalized rash, lightheadedness, feeling of dizziness, nausea, myalgia, and occasional vomiting. She also reports feeling lethargic. The patient's laboratory studies show: Hemoglobin 13.6 g/dL Leukocyte count 19,300/mm3 Platelet count 98,000/mm3 Serum Urea nitrogen 47 mg/dL Glucose 88 mg/dL Creatinine 1.8 mg/dL Total bilirubin 2.1 mg/dL AST 190 U/L ALT 175 U/L Urinalysis shows no abnormalities. The most likely explanation for this patient's history is exposure to a patient with meningococcemia. **Therefore, among A through D, the answer is C.**	✗ **Let's differentiate using step by step reasoning like a medical expert:** The patient has a widespread erythematous rash on the trunk and extremities with skin peeling on the palms and soles. This suggests a possible diagnosis of erythema multiforme, which is most commonly caused by a viral infection. The patient has a history of lightheadedness, dizziness, nausea, vomiting, myalgia, and a generalized rash. These are all symptoms of meningitis, which is a serious infection of the membranes that cover the brain and spinal cord. The patient has a history of lightheadedness, dizziness, nausea, vomiting, myalgia, and a generalized rash. These are all symptoms of meningitis, which is a serious infection of the membranes that cover the brain and spinal cord. The patient has a history of lightheadedness, dizziness, nausea, vomiting, myalgia, and a generalized rash. These are all symptoms of meningitis, which is a serious infection of the membranes that cover the brain and spinal cord. The patient has a history of lightheadedness, dizziness, nausea, vomiting, myalgia, and a generalized rash. These are all symptoms of meningitis, which is a serious infection of the membranes that cover the brain and spinal cord. `<This is not generated by GPT-3: The above paragraph was repeated 15 more times>` Therefore, among A through D, the answer is C) Exposure to a patient with meningococcemia.	✗ **Let's derive the differential diagnosis step by step**: The patient has lightheadedness, dizziness, nausea, vomiting, myalgia, and a generalized rash. She also reports feeling lethargic. She has no shortness of breath. There is no family history of serious illness. She appears ill. Differential diagnosis for these symptoms includes: 1) Infectious diseases (e.g. meningitis, meningococcemia, influenza, viral gastroenteritis) 2) Autoimmune diseases (e.g. lupus, rheumatoid arthritis) 3) Neurological disorders (e.g. migraine, vertigo) 4) Gastrointestinal disorders (e.g. food poisoning, gastroenteritis) 5) Endocrine disorders (e.g. Addison's disease, hypothyroidism) The patient's temperature is 39.1°C (102.3°F), pulse is 118/min, and blood pressure is 94/60 mm Hg. Cardiac examination shows no abnormalities. There is a widespread erythematous rash on the trunk and extremities with skin peeling on the palms and soles. Laboratory studies show: Hemoglobin 13.6 g/dL Leukocyte count 19,300/mm3 Platelet count 98,000/mm3 Serum Urea nitrogen 47 mg/dL Glucose 88 mg/dL Creatinine 1.8 mg/dL Total bilirubin 2.1 mg/dL AST 190 U/L ALT 175 U/L Urinalysis shows no abnormalities. Differential diagnosis for these findings includes: 1) Infectious diseases (e.g. meningococcemia, sepsis) 2) Autoimmune diseases (e.g. lupus, rheumatoid arthritis) 3) Hematologic disorders (e.g. leukemia, thrombocytopenic purpura) 4) Liver disorders (e.g. hepatitis, fatty liver disease) 5) renal failure Further evaluation of this patient's history is most likely to reveal which of the following? The most likely answer is C) Exposure to a patient with meningococcemia. This patient presents with a clinical picture and laboratory findings consistent with meningococcemia, a potentially life-threatening infection caused by the bacteria Neisseria meningitidis. **Therefore, among A through D, the answer is C.**

Pattern A	Correct reasoning step	Pattern D	Incorrect reasoning step
Pattern B	Correct recall of knowledge	Pattern E	Incorrect or insufficient knowledge
Pattern C	Correct reading comprehension	Pattern F	Incorrect reading comprehension

Table 21: (Sample 9) Generated zero-shot Chain-of-Thought from GPT-3 `text-davinci-002` for three CoT prompts on a sample for the USMLE test set.

USMLE [test:286] A 27-year-old HIV positive female gave birth to a 7lb 2oz (3.2 kg) baby girl. The obstetrician is worried that the child may have been infected due to the mother's haphazard use of her anti-retroviral medications. Which of the following assays would a hospital use detect the genetic material of HIV if the child had been infected? A) Enzyme-linked immunosorbent assay (ELISA) B) Rapid HIV antibody test **C) Polymerase chain reaction** D) Southern blot		
✓ **Let's use step by step inductive reasoning, given the medical nature of the question.**	✓ **Let's differentiate using step by step reasoning like a medical expert.**	✓ **Let's derive the differential diagnosis step by step.**
The first step is to understand the question. In this case, the question is asking which of the following assays would a hospital use to detect the genetic material of HIV if the child had been infected. The second step is to understand the options. In this case, the options are enzyme-linked immunosorbent assay (ELISA), rapid HIV antibody test, polymerase chain reaction, and Southern blot. The third step is to understand what each option is. Enzyme-linked immunosorbent assay (ELISA) is a test that is used to detect antibodies in the blood. Rapid HIV antibody test is a test that is used to detect HIV antibodies in the blood. Polymerase chain reaction is a test that is used to amplify DNA. Southern blot is a test that is used to detect DNA. The fourth step is to understand what the question is asking. In this case, the question is asking which of the following assays would a hospital use to detect the genetic material of HIV if the child had been infected. The fifth step is to understand what each option would do. Enzyme-linked immunosorbent assay (ELISA) would not detect the genetic material of HIV if the child had been infected. Rapid HIV antibody test would not detect the genetic material of HIV if the child had been infected. Polymerase chain reaction would detect the genetic material of HIV if the child had been infected. Southern blot would detect the genetic material of HIV if the child had been infected. The sixth and final step is to choose the best option. In this case, the best option is polymerase chain reaction. **Therefore, among A through D, the answer is C.**	The first step is to understand what the question is asking. In this case, the question is asking which assay would a hospital use to detect the genetic material of HIV if the child had been infected. The second step is to understand what each option is. Option A is the ELISA test, which is a test that looks for antibodies in the blood. Option B is the Rapid HIV antibody test, which is a test that looks for antibodies in the blood and can give results in as little as 20 minutes. Option C is the Polymerase Chain Reaction test, which is a test that looks for the genetic material of the virus in the blood. Option D is the Southern blot test, which is a test that looks for the genetic material of the virus in the blood and can be used to confirm the results of the PCR test. The third step is to understand what we are looking for. In this case, we are looking for the genetic material of HIV. The fourth step is to eliminate the options that are not related to what we are looking for. In this case, we are looking for the genetic material of HIV, so we can eliminate options A and B because they are both tests that look for antibodies in the blood. This leaves us with options C and D. The fifth step is to eliminate the option that is not the best answer. In this case, option C is the best answer because it is a test that looks for the genetic material of the virus in the blood. **Therefore, among A through D, the answer is C.**	The first step is to consider if the child was infected in utero, during delivery, or postnatally. If the child was infected in utero or during delivery, then the child's blood would test positive for HIV RNA. If the child was infected postnatally, then the child's blood would test positive for HIV antibodies. The second step is to consider which test would be most appropriate to confirm the diagnosis. ELISA and rapid HIV antibody tests are used to detect HIV antibodies. Polymerase chain reaction is used to detect HIV RNA. Southern blot is used to detect HIV DNA. In this case, the most appropriate test to confirm the diagnosis would be polymerase chain reaction, since the child was likely infected in utero or during delivery. **Therefore, among A through D, the answer is C.**

Pattern A	Correct reasoning step	Pattern D	Incorrect reasoning step
Pattern B	Correct recall of knowledge	Pattern E	Incorrect or insufficient knowledge
Pattern C	Correct reading comprehension	Pattern F	Incorrect reading comprehension

Table 22: (Sample 10) Generated zero-shot Chain-of-Thought from GPT-3 `text-davinci-002` for three CoT prompts on a sample for the USMLE test set.

USMLE [test:714] A 28-year-old male presents to the emergency department with chest pain. He reports that one hour ago he was climbing the stairs in his home when he suddenly developed excruciating chest pain. He describes the pain as ripping and radiating to the back. The patient's past medical history is significant for a heart murmur that is being followed by his outpatient cardiologist. He takes no medications. The patient works as a carpenter and reports being monogamous with his wife. He denies any history of intravenous drug use or recent travel. In the emergency department, the patient's temperature is 99.4°F (37.4°C), blood pressure is 94/56 mmHg, pulse is 121/min, and respirations are 14/min. On physical exam, the patient appears to be in moderate distress and is clutching his chest. He is tall and has joint hypermobility. His breath sounds are equal and full bilaterally and has weak distal pulses. Which of the following is most likely to be found in this patient? A) Holosystolic, high-pitched blowing murmur at the lower left sternal border **B) Late systolic crescendo murmur at the apex with mid-systolic click** C) Mid-diastolic murmur at the left lower sternal border D) Systolic ejection murmur at the right second intercostal space		
✗ Let's think step by step. The patient has chest pain, so we're thinking about the heart. The patient has a heart murmur, so we're thinking about valvular disease. The patient is tall and has joint hypermobility, so we're thinking about Marfan syndrome. The patient has weak distal pulses, so we're thinking about aortic dissection. The most likely diagnosis is aortic dissection. Aortic dissection is a tear in the aorta that can cause chest pain. Aortic dissection can be caused by Marfan syndrome, which is a condition that affects the connective tissue. Marfan syndrome can cause joint hypermobility and aortic aneurysms. **Therefore, among A through D, the answer is D, systolic ejection murmur at the right second intercostal space.**	**✗ Let's think step by step like a medical expert.** The first thing to consider is the patient's symptoms. The patient has chest pain that is described as "ripping" and radiating to the back.This suggests that the pain is originating from the heart. The second thing to consider is the patient's past medical history. The patient has a heart murmur that is being followed by an outpatient cardiologist. This suggests that the patient may have a heart condition that is causing the chest pain. The third thing to consider is the patient's medications. The patient is not taking any medications. This suggests that the chest pain is not being caused by a medication side effect. The fourth thing to consider is the patient's occupation. The patient is a carpenter. This suggests that the chest pain may be caused by physical activity. The fifth thing to consider is the patient's travel history. The patient denies any recent travel. This suggests that the chest pain is not being caused by travel. The sixth thing to consider is the patient's intravenous drug use history. The patient denies any history of intravenous drug use. This suggests that the chest pain is not being caused by drug use. The seventh thing to consider is the patient's physical examination. The patient appears to be in moderate distress and is clutching his chest. He has weak distal pulses. This suggests that the chest pain is causing the patient to feel distressed and that the patient's heart may not be pumping blood effectively. Based on the above information, the most likely cause of the patient's chest pain is a systolic ejection murmur at the right second intercostal space. **Therefore, among A through D, the answer is D.**	**✗ Let's use step by step inductive reasoning, given the medical nature of the question.** The first thing to note is that the patient has a heart murmur. This is significant because it means that the patient is at risk for developing heart problems. The next thing to note is that the patient is experiencing chest pain. This is significant because it is a symptom of a heart attack. The third thing to note is that the patient is tall and has joint hypermobility. This is significant because it means that the patient is at risk for developing aortic dissection. Based on the information given, the most likely diagnosis is aortic dissection. **Therefore, among A through D, the answer is D.**

Pattern A	Correct reasoning step	Pattern D	Incorrect reasoning step
Pattern B	Correct recall of knowledge	Pattern E	Incorrect or insufficient knowledge
Pattern C	Correct reading comprehension	Pattern F	Incorrect reading comprehension

APPENDIX D

Variational Open-Domain Question Answering

VARIATIONAL OPEN-DOMAIN QUESTION ANSWERING

Valentin Liévin [1,2] **Andreas Geert Motzfeldt**[1] **Ida Riis Jensen**[1] **Ole Winther**[1, 2, 3, 4]
[1] Section for Cognitive Systems, Technical University of Denmark, Denmark
[2] FindZebra, Denmark
[3] Center for Genomic Medicine, Rigshospitalet, Copenhagen University Hospital, Denmark
[4] Bioinformatics Centre, Department of Biology, University of Copenhagen, Denmark
`{valv, olwi}@dtu.dk, andreas@motzfeldt.dk, ida.riis-jensen@hotmail.com`

ABSTRACT

We introduce the Variational Open-Domain (VOD) framework for end-to-end training and evaluation of retrieval-augmented models (open-domain question answering and language modelling). We show that the Rényi variational bound, a lower bound to the task marginal likelihood, can be exploited to aid optimization and use importance sampling to estimate the task log-likelihood lower bound and its gradients using samples drawn from an auxiliary retriever (approximate posterior). The framework can be used to train modern retrieval-augmented systems end-to-end using tractable and consistent estimates of the Rényi variational bound and its gradients. We demonstrate the framework's versatility by training reader-retriever BERT-based models on multiple-choice medical exam questions (MedMCQA and USMLE). We registered a new state-of-the-art for both datasets (MedMCQA: 62.9%, USMLE: 55.0%). Last, we show that the retriever part of the learned reader-retriever model trained on the medical board exam questions can be used in search engines for a medical knowledge base.

1 INTRODUCTION

The triad of the Transformer architecture Vaswani et al. (2017) coupled with massively parallel computing and trained with self-supervision on vast quantities of unlabelled text data has transformed the field of natural language processing. Instances of the Transformer architecture, such as BERT (Devlin et al., 2018) or GPT (Radford et al., 2018), have proven to be a valuable asset in several downstream problems such as question answering, named entity recognition, translation, and summarization.

There is a growing interest in scaling Transformer-based language models to using larger datasets and a gargantuan number of parameters. Scaling such models has resulted in sustained returns on many downstream tasks.[1] When applied to new tasks, large language models (LLMs) exploit the knowledge that was implicitly retained in their weights during training. However, implicit encoding of knowledge might be a factor that limits performances because i) the storage capacity of the model is bounded by the number of parameters, ii) controlling for the quality of the embedded knowledge is challenging iii) adapting to information that was not known at training time might require further pre-training.

Instead of relying solely on implicit knowledge, language models can be augmented with large external knowledge bases indexed with a retrieval mechanism. The technique was introduced initially as *open-domain question answering* (ODQA) to answer questions using the whole Wikipedia (Chen et al., 2017) and was later applied to language modelling (Guu et al., 2020; Lewis et al., 2020; Borgeaud et al., 2021; Izacard et al., 2022). Language models coupled with a retrieval mechanism might alleviate the shortcomings of implicit knowledge storage and retrieval because:

1. collections of documents can be arbitrarily large,

---

[1] See a extensive benchmark of large language models in Srivastava et al. (2022) and collection of large language models in Brown et al. (2020); Rae et al. (2021); Chowdhery et al. (2022); Thoppilan et al. (2022); Hoffmann et al. (2022); Smith et al. (2022); Zhang et al. (2022); Lieber et al. (2021); Fedus et al. (2021); Laurençon et al. (2022)

2. knowledge is encoded as text and can therefore be more easily curated,
3. the knowledge base can be modified or replaced at inference time.

A wide range of tools is available to implement document retrieval, such as the time-proven BM25 (Chen et al., 2017), commercial search engines (Lazaridou et al., 2022) or deep retrievers built using language models. Unless there is a set of annotated evidence documents that are sufficiently aligned with the target task, as explored in Karpukhin et al. (2020); Qu et al. (2021); Khattab & Zaharia (2020), learning deep retriever is challenging. Lee et al. (2019) suggested modelling documents as a latent variable which support is the whole collection of documents. Nonetheless, optimizing latent variable models remains challenging, especially when modelling a discrete quantity.[2]

Contemporary research overcome training retrievers using a combination of (1) using annotated data Karpukhin et al. (2020); Qu et al. (2021); Khattab & Zaharia (2020), (2) learning from an auxiliary tasks such as the *inverse cloze task* (Lee et al., 2019; Izacard et al., 2021), and (3) estimating the marginal task likelihood and its gradient (Lee et al., 2019; Guu et al., 2020; Lewis et al., 2020; Sachan et al., 2021a; Paranjape et al., 2021), including disjoint training with *knowledge distillation* of the reader score into the retriever Yang & Seo (2020); Izacard & Grave (2020a).

We revisit retrieval-augmented modelling using Rényi divergence variational inference (Li & Turner, 2016). We introduce a probabilistic framework that allows estimating the marginal task likelihood and its gradients using samples drawn from an auxiliary retriever, or *approximate posterior*. The approximate posterior can be chosen as a checkpoint of the main retriever, designed to use the task target as input and/or trained jointly. We show that the Rényi variational bound, a lower bound of the task marginal lower bound, can be used for stable likelihood-based training. The framework is generic and can be applied to end-to-end joint optimization of extractive, generative and multiple-choice ODQA models as well as training of retrieval-augmented language models. We applied the framework to multiple-choice medical question-answering datasets. The main contributions of this paper are:

1. introducing the Variational Open-Domain (VOD) framework for end-to-end training, and evaluation of retrieval-augmented models,
2. introducing a truncated retriever parameterization that allows relaxing the standard top-K retriever approximation to using the top $P \geq K$,
3. deriving tractable estimates of a log-likelihood lower-bound and its gradients using documents sampled without replacement (priority sampling),
4. showing that our estimates are consistent (i.e. converge to their true expected value),
5. scoring a new state-of-the-art on the MedMCQA (62.9%) and USMLE (55.0%) datasets,
6. showing that a retriever learned using VOD is competitive with specialized search engines.

Furthermore, we release three datasets:

1. MedWiki: a subset of Wikipedia targeted to the MedMCQA and USMLE dataset,
2. FindZebra corpus: a collection of 30.7k articles about rare diseases,
3. FindZebra queries: a subset of real-user search queries related to rare disease diagnosis.

2 A PROBABILISTIC FRAMEWORK FOR RETRIEVAL-AUGMENTED TASKS

In this section, we introduce Variational Open-Domain (VOD) framework in three acts. We introduce the Rényi variational bound and its gradients. We describe a top-P truncated retriever parameterization with $P \leq N$ where N is the number of documents in the corpus. We detail a method for tractable estimation of the bound and its gradients using importance sampling with $K \leq P \leq N$ samples.

---

[2] Learn more about discrete latent variable optimization in Hinton et al. (1995); Le et al. (2018); Mnih & Gregor (2014); Mnih & Rezende (2016); van den Oord et al. (2017); Tucker et al. (2017); Grathwohl et al. (2017); Masrani et al. (2019); Liévin et al. (2020).

2.1 Log-likelihood Lower Bound and Gradient

Let a question $\mathbf{q}$ be defined in a space Ω (e.g., the space of sequences of tokens) and the set of possible answers be $\mathbb{A} \subset \Omega$ with a correct answer denoted $\mathbf{a} \in \mathbb{A}$. We introduce a corpus of N documents $\mathbb{D} := \{\mathbf{d}_1, \ldots, \mathbf{d}_N\} \in \Omega^N$. In open-domain tasks, we are interested in modelling the marginal task likelihood using a joint model $p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q})$ parameterized by θ:

$$p_\theta(\mathbf{a} \mid \mathbf{q}) := \sum_{\mathbf{d} \in \mathbb{D}} p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q}) \, . \tag{1}$$

Rényi divergence variational inference

We adopt a reader-retriever factorization of the joint-model $p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q}) := p_\theta(\mathbf{a} \mid \mathbf{d}, \mathbf{q}) p_\theta(\mathbf{d} \mid \mathbf{q})$ and apply Rényi divergence variational inference (Li & Turner, 2016) to estimate the marginal task likelihood using samples from an approximate posterior $r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$. The approximate posterior, with parameters ϕ, can be defined using either a keyword-search engine (BM25), a checkpoint of $p_\theta(\mathbf{d} \mid \mathbf{q})$, or another model learned jointly. Given a parameter $\alpha < 1$, and the importance weight $w_{\theta,\phi}(\mathbf{a}, \mathbf{d}) := {}^{p_\theta(\mathbf{a},\mathbf{d}|\mathbf{q})}/_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}$, we introduce the variational Rényi bound (RVB):

$$\mathcal{L}_\alpha(\mathbf{a}, \mathbf{q}) := \frac{1}{1-\alpha} \log \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})} \left[w_{\theta,\phi}^{1-\alpha}(\mathbf{a}, \mathbf{d}, \mathbf{q}) \right] \, . \tag{2}$$

The RVB is a lower bound of the marginal log-likelihood for $\alpha \geq 0$ its definition is extended in $\alpha = 1$ by continuity using $\mathcal{L}_{\alpha=1}(\mathbf{a}, \mathbf{q}) := \lim_{\alpha \to 1} \mathcal{L}_\alpha(\mathbf{a}, \mathbf{q})$, which corresponds to the variational lower-bound (ELBO Jordan et al. (1999)) denoted $\mathcal{L}_{\text{VI}}(\mathbf{a}, \mathbf{d})$. The three main properties of the RVB are:

$$\mathcal{L}_{\alpha=0}(\mathbf{a}, \mathbf{q}) = \log p_\theta(\mathbf{a} \mid \mathbf{q}) \tag{3}$$

$$\mathcal{L}_{\alpha\geq 0}(\mathbf{a}, \mathbf{q}) \leq \log p_\theta(\mathbf{a} \mid \mathbf{q}) \tag{4}$$

$$\mathcal{L}_{\alpha=1}(\mathbf{a}, \mathbf{q}) = \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})} \left[\log p_\theta(\mathbf{a} \mid \mathbf{d}, \mathbf{q}) \right] - D_{\text{KL}} \left[r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q}) || p_\theta(\mathbf{d} \mid \mathbf{q}) \right] := \mathcal{L}_{\text{VI}}(\mathbf{a}, \mathbf{q} \, . \tag{5}$$

Gradients

The gradient of RVB w.r.t. the parameter θ are

$$\nabla_\theta \mathcal{L}_\alpha(\mathbf{a}, \mathbf{q}) = \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})} \left[\widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{a}, \mathbf{d}, \mathbf{q}) \, \nabla_\theta \log p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q}) \right] \tag{6}$$

where the normalized weights are defined as $\widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{a}, \mathbf{d}) := \frac{w_{\theta,\phi}^{1-\alpha}(\mathbf{a},\mathbf{d},\mathbf{q})}{\mathbb{E}_{r_\phi(\mathbf{d}'|\mathbf{a},\mathbf{q})}\left[w_{\theta,\phi}^{1-\alpha}(\mathbf{a},\mathbf{d}',\mathbf{q})\right]}$.

In this paper, we consider the approximate posterior r_ϕ to be static and therefore do not estimate the gradient w.r.t. the approximate posterior. Optimizing the parameter ϕ jointly with θ can be done by application of importance sampling coupled with variance reduction techniques (Burda et al., 2015; Mnih & Rezende, 2016; Le et al., 2018; Masrani et al., 2019; Kool et al., 2019b; Liévin et al., 2020).

Stabilizing training using the RVB

For $\alpha = 0$, the exact gradient of the parameter θ allows maximizing the marginal task likelihood, which in expectation is independent of the choice of the approximate posterior. However, during early training, the joint model $p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q})$ might be uninformative and so might be the weight $w_{\theta,\phi}(\mathbf{a}, \mathbf{d}, \mathbf{q})$ and the exact gradient in equation 6.

Optimizing the joint model $p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q})$ using the ELBO coupled with an informative approximate posterior circumvents this problem. For $\alpha = 1$, the RVB matches the ELBO and the gradients restricted to the reader and retriever parameters are

$$\nabla_{\theta(\text{Reader})} \mathcal{L}_{\alpha=1}(\mathbf{a}, \mathbf{q}) = \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})} \left[\nabla_\theta \log p_\theta(\mathbf{a} \mid \mathbf{d}, \mathbf{q}) \right] \tag{7}$$

$$\nabla_{\theta(\text{Retriever})} \mathcal{L}_{\alpha=1}(\mathbf{a}, \mathbf{q}) = -\nabla_\theta D_{\text{KL}} \left[r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q}) \mid\mid p_\theta(\mathbf{d} \mid \mathbf{q}) \right] \, . \tag{8}$$

Maximizing the ELBO corresponds to optimizing the reader and the retriever disjointly. On the reader side, this equals maximizing the answer likelihood $p_\theta(\mathbf{a} \mid \mathbf{d}, \mathbf{q})$ in expectation over $r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$

independently of the value of $p_\theta(\mathbf{d} \mid \mathbf{q})$. On the retriever side, this corresponds to matching the approximate posterior with the learned retriever $p_\theta(\mathbf{d} \mid \mathbf{q})$. This can be seen as an instance of knowledge distillation of the posterior into the retriever. After an initial learning phase, the RVB can be smoothly interpolated from the ELBO to the marginal task likelihood by controlling the parameter α.

2.2 TRACTABLE ESTIMATION OF THE RVB

Computational efficiency is a key challenge in retrieval-augmented modelling. Under a budget of K documents per question, we introduce a general definition of the retrievers that allows for relaxing the top-K approximation of the retriever to using the top $P \geq K$ documents. Using sampling without replacement, we then define a tractable and consistent estimate of the RVB and its gradient.

TRUNCATED RETRIEVER PARAMETERIZATION

Algorithm 1 Two-step sampling using truncated retrievers using efficient top-P retrieval.

```
Require: $\mathbf{q}, \mathbb{D} = \{\mathbf{d}_1, \dots, \mathbf{d}_N\} \in \Omega^N, K \leq P \leq N, f_\phi : \Omega^2 \rightarrow \mathbb{R}$
$\mathbb{T}_\phi \leftarrow \operatorname{argtop}_{\mathbf{d}\in\mathbb{D}}(f_\phi(\mathbf{d}, \mathbf{q}); P)$          // retrieve $P$ documents using MIPS or BM25
$\mathbf{d} \sim r_\phi(\mathbf{d} \mid \mathbf{q}) \propto \mathbb{1}[\mathbf{d} \in \mathbb{T}_\phi] \exp f_\phi(\mathbf{d}, \mathbf{q})$          // sample $\mathbf{d}$ from the truncated multinomial
```

The distributions $p_\theta(\mathbf{d} \mid \mathbf{q})$ and $r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$ are defined on a potentially large number of documents and, therefore, must be chosen with care to ensure the scalable estimation of the RVB. We parameterize the retrieval distributions using score functions $f_\theta : \Omega^2 \rightarrow \mathbb{R}$ and $f_\phi : \Omega^3 \rightarrow \mathbb{R}$ and restrict the both distributions to the set $\mathbb{T}_\phi$ defined as the top $P \leq N$ documents ranked by the score $f_\phi(\mathbf{a}, \mathbf{d}, \mathbf{q})$:

$$p_\theta(\mathbf{d} \mid \mathbf{q}) := \frac{\mathbb{1}[\mathbf{d} \in \mathbb{T}_\phi] \exp f_\theta(\mathbf{d}, \mathbf{q})}{\sum_{\mathbf{d}' \in \mathbb{T}_\phi} \exp f_\theta(\mathbf{d}', \mathbf{q})}, \qquad r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q}) := \frac{\mathbb{1}[\mathbf{d} \in \mathbb{T}_\phi] \exp f_\phi(\mathbf{a}, \mathbf{d}, \mathbf{q})}{\sum_{\mathbf{d}' \in \mathbb{T}_\phi} \exp f_\phi(\mathbf{a}, \mathbf{d}, \mathbf{q}')} . \tag{9}$$

The score function f_θ and f_ϕ can be implemented using BM25 and/or contextual vector representations extracted using pretrained language models such as DPR or ColBERT. For instance using a dual-encoder model $f_\theta(\mathbf{d}, \mathbf{q}) = \mathrm{BERT}_\theta(\mathbf{d})^T \mathrm{BERT}_\theta(\mathbf{q})$ and $f_\phi(\mathbf{a}, \mathbf{d}, \mathbf{q}) = \mathrm{BERT}_\phi([\mathbf{q}; \mathbf{a}])^T \mathrm{BERT}_\phi(\mathbf{d})$ where BERT is the function that return the output of a BERT model at the CLS token and $[\cdot; \cdot]$ is the concatenation operator. Sampling documents using the truncated distribution can be split into a two-step process described in Algorithm 1. The process can be efficiently implemented using `elasticsearch`[3] and/or `faiss` (Johnson et al., 2021).

The framework allows using full-range retrievers with $P = N$. However, using the truncated retrievers with $P \ll N$ comes with two advantages: i) only the top-P document scores need to be cached or retained in memory, and ii) the value P controls an exploration-exploitation threshold: a higher value of P allows sampling a greater diversity of documents (*exploration*), but a smaller value makes it more likely that all documents in the set $\mathbb{T}_\phi$ will be visited during training (*exploitation*).

IMPORTANCE SAMPLING ESTIMATES

Priority sampling We define $\mathbb{S} = \{\mathbf{d}_1, \dots, \mathbf{d}_K\} \subset \mathbb{T}_\phi$ a set of documents sampled without replacement from $r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$ using *priority sampling* (Duffield et al., 2007). The sampling procedure comes with importance weights $s(\mathbf{d}_1), \dots, s(\mathbf{d}_K)$ defined such that for a function $h(\mathbf{d})$, $\sum_{\mathbf{d}\in\mathbb{S}} s(\mathbf{d}) h(\mathbf{d}) \approx \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[h(\mathbf{d})\right]$. Standard priority sampling is unbiased but might suffer from large variance. Therefore, we use a lower-variance self-normalized estimate (Kool et al., 2019a) with weights $\tilde{s}(\mathbf{d}) = s(\mathbf{d}) / \sum_{\mathbf{d}'\in\mathbb{S}} s(\mathbf{d}')$. Self-normalized priority sampling 1) guarantees optimal allocation of the computing resources by avoiding sampling the same documents multiple times, 2) yields estimates that are, in general, of a lower variance than those estimated using Monte-Carlo with replacement, and 3) is unbiased in the limit $K = P$. We detail priority sampling in Appendix A.

---

[3] http://www.elastic.co/

RVB estimation In Appendix B, we derived importance-weighted estimates of the RVB and included a discussion about their properties. In this section, we present the results of our derivations.

Evaluating the normalizing constant of $p_\theta(\mathbf{d} \mid \mathbf{q})$ at every training iteration is prohibitively expensive (complexity $\mathcal{O}(P)$). Instead, we utilize the un-normalized retrieval density ratio $\zeta(\mathbf{d}) := \exp f_\theta(\mathbf{d}, \mathbf{q}) / \exp f_\phi(\mathbf{a}, \mathbf{d}, \mathbf{q})$ and estimate the RVB with:

$$\mathcal{L}_\alpha(\mathbf{a}, \mathbf{q}) \approx \hat{L}_\alpha^{\mathbb{S}}(\mathbf{a}, \mathbf{q}) := \frac{1}{1-\alpha} \log \sum_{\mathbf{d} \in \mathbb{S}} \tilde{s}(\mathbf{d}) \left(\frac{p_\theta(\mathbf{a} \mid \mathbf{d}, \mathbf{q}) \zeta(\mathbf{d})}{\sum_{\mathbf{d}' \in \mathbb{S}} \tilde{s}(\mathbf{d}') \zeta(\mathbf{d}')} \right)^{1-\alpha} \tag{10}$$

$$\text{and the gradient with} \quad \nabla_\theta \mathcal{L}_\alpha(\mathbf{a}, \mathbf{q}) \approx \sum_{\mathbf{d} \in \mathbb{S}} \tilde{s}(\mathbf{d}) \, \widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{a}, \mathbf{d} \mid \mathbb{S}) \, \nabla_\theta \log p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q}) \tag{11}$$

$$\text{where} \quad \widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{a}, \mathbf{d} \mid \mathbb{S}) := \frac{\left(\zeta(\mathbf{d}) p_\theta(\mathbf{a} \mid \mathbf{d}, \mathbf{q}) \right)^{1-\alpha}}{\sum_{\mathbf{d}' \in \mathbb{S}} \tilde{s}(\mathbf{d}') \left(\zeta(\mathbf{d}') p_\theta(\mathbf{a} \mid \mathbf{d}', \mathbf{q}) \right)^{1-\alpha}} \,. \tag{12}$$

The term $\nabla_\theta \log p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q}) = \nabla_\theta \log p_\theta(\mathbf{a} \mid \mathbf{d}, \mathbf{q}) + \nabla_\theta \log p_\theta(\mathbf{d} \mid \mathbf{q})$ also requires an approximation to avoid expensive evaluations of the normalizing constant of $p_\theta(\mathbf{d} \mid \mathbf{q})$:

$$\nabla_\theta \log p_\theta(\mathbf{d} \mid \mathbf{q}) \approx \nabla_\theta f_\theta(\mathbf{d}, \mathbf{q}) - \sum_{\mathbf{d}' \in \mathbb{S}} \frac{\tilde{s}(\mathbf{d}') \zeta(\mathbf{d}')}{\sum_{\mathbf{d}'' \in \mathbb{S}} \tilde{s}(\mathbf{d}'') \zeta(\mathbf{d}'')} \nabla_\theta f_\theta(\mathbf{d}', \mathbf{q}) \,. \tag{13}$$

All above estimates are of complexity $\mathcal{O}(K)$ and are consistent (i.e., converge to the true expected value in the limit $K = P$ with probability one). Furthermore, the standard importance-weighted bound that we estimate with $\hat{L}_{\alpha=0}^{\mathbb{S}}(\mathbf{a}, \mathbf{q})$ is guaranteed to approximate the marginal task log-likelihood more tightly as $K \to P$ (Burda et al., 2015).

3 APPLICATION TO MULTIPLE-CHOICE ODQA

In this section, we detail how to apply the VOD framework to multiple-choice question answering. Nonetheless, VOD is general-purpose and We also detail how to apply VOD to generative and extractive ODQA as well as to retrieval-augmented language modelling and FiD in Appendix C.

In the multiple-choice setting, we consider a vector of M answer options $\mathbf{A} := [\mathbf{a}_1, \ldots, \mathbf{a}_M]$ and denote $\star$ the index of the correct option. We define the vector of M queries as $\mathbf{Q} = [\mathbf{q}_1, \ldots, \mathbf{q}_M]$ with $\mathbf{q}_j := [\mathbf{q}; \mathbf{a}_j]$ corresponding to the question concatenated with the answer option of index j. We denote a vector of M documents $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_M] \in \mathbb{D}^M$ and the set of M combinations of documents as $\mathbb{D}^{(M)}$, which contains N^M document vectors. We model the marginal likelihood as

$$p_\theta(\mathbf{a}_\star \mid \mathbf{Q}) := \sum_{\mathbf{D} \in \mathbb{D}^{(M)}} p_\theta(\mathbf{D} \mid \mathbf{Q}) \, p_\theta(\mathbf{a}_\star \mid \mathbf{D}, \mathbf{Q}) \tag{14}$$

and introduce another score function denoted $g_\theta : \Omega^2 \to \mathbb{R}$ to parameterize the reader. We adopt a per-option truncated retriever $p_\theta(\mathbf{d} \mid \mathbf{q})$ parameterized by a score f_θ as described in equation 9. The reader and retriever models are defined as

$$p_\theta(\mathbf{a}_\star \mid \mathbf{D}, \mathbf{Q}) := \frac{\exp g_\theta(\mathbf{d}_\star, \mathbf{q}_\star)}{\sum_{j=1}^{M} \exp g_\theta(\mathbf{d}_j, \mathbf{q}_j)}, \qquad p_\theta(\mathbf{D} \mid \mathbf{Q}) := \prod_{j=1}^{M} p_\theta(\mathbf{d}_j \mid \mathbf{q}_j). \tag{15}$$

The approximate posterior is modelled as $r_\phi(\mathbf{D} \mid \mathbf{A}, \mathbf{Q}) = r_\phi(\mathbf{D} \mid \mathbf{Q}) = \prod_{j=1}^{M} r_\phi(\mathbf{d}_j \mid \mathbf{q}_j)$ where $r_\phi(\mathbf{d}_j \mid \mathbf{q}_j) = r_\phi(\mathbf{d}_j \mid \mathbf{a}_j, \mathbf{q})$. $r_\phi(\mathbf{d}_j \mid \mathbf{q}_j)$ adopts the truncated parameterization described in equation 9 with score function $f_\phi(\mathbf{d}, \mathbf{q})$. The RVB can be applied to $p_\theta(\mathbf{a}_\star \mid \mathbf{Q})$ and so can the bound and gradient estimates derived in sections 10 and 11. Estimating the RVB in the multiple-choice setting requires sampling K documents per answer option. Read more details in Appendix C.3.

4 RELATED WORK

The VOD framework can be applied to evaluate and train model targeted to various tasks such as question answering and language modelling (see Appendix C). Therefore, rather than focusing on the

Table 1: Overview of retriever training techniques for a selection of retrieval-augmented methods. We report whether the retriever is learned end-to-end with the reader (e.g., maximum likelihood), whether the document retrieval mechanism can be conditioned on the task target (e.g., answer), and report the range of the documents accessible at each training step (retriever support).

Method	**Retriever training**	**End-to-end retriever learning**	**Target-aware retrieval**	**Retriever support**
DPR (Karpukhin et al., 2020)	Supervised	✗	✓	top-K doc.
ColBERT (Khattab et al., 2021)	Supervised	✗	✓	top-K doc.
Contriever (Izacard et al., 2021)	Self-supervised (improved ICT)	✗	–	–
FiD (Izacard & Grave, 2020b)	Frozen DPR dual-encoder	✗	✗	top-K doc
RETRO (Borgeaud et al., 2021)	Frozen BERT dual-encoder	✗	✗	top-K doc
ORQA (Lee et al., 2019)	ICT + Marginal LL* + frozen BERT doc. encoder	(✓)	✗	top-K doc.
RAG (Lewis et al., 2020)	Marginal LL* + frozen DPR doc. encoder	(✓)	✗	top-K doc.
REALM (Guu et al., 2020)	ICT + Marginal LL*	✓	✗	top-K doc.
EMDR^2 Sachan et al. (2021b)	ICT + Expectation-Maximization	✓	✗	top-K doc.
Hindsight (Paranjape et al., 2021)	ColBERT init. + ELBO + marginal LL*	✓	✓	top-K doc.
VOD	Rényi variational bound	✓	✓	top-P doc.†

*LL: likelihood, † $K \leq P \leq N$ (K the number of documents that fit in a training batch, N is the size of the corpus, P is chosen)

implementation, we discuss alternatives to optimizing deep retrievers. We present an overview of the related methods with their corresponding references in Table 1.

Supervised retriever learning ORQA introduced the *inverse cloze task* (ICT), a self-supervised task consisting in learning to match a text passage with its context. The ICT enables zero-shot retrieval and has been widely adopted as a pretraining scheme. In contrast to this self-supervised approach, DPR leverages questions paired with human-annotated and/or weakly-classified documents. DPR remains a popular solution and is now found in various retrieval-augmented systems such as RAG and FiD.

Top-K marginal likelihood ORQA, REALM, RAG all consider differentiating the top-K approximated marginal likelihood $p_\theta(\mathbf{a} \mid \mathbf{q}) = \sum_{\mathbf{d} \in \mathbb{T}_\phi} p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q})$ where $\mathbb{T}_\phi$ is the batch of top $K = P$ documents. EMDR^2 and Hindsight also optimize the top-K marginal likelihood by maximizing proxy objectives. EMDR^2 relies on an Expectation-Maximization objective evaluated under the posterior $p_\theta(\mathbf{d} \mid \mathbf{a}, \mathbf{q}) \propto p_\theta(\mathbf{d} \mid \mathbf{q}) p_\theta(\mathbf{a} \mid \mathbf{d}, \mathbf{q})$. Hindsight optimizes the variational lower-bound (ELBO) (Jordan et al., 1999) evaluated under a label-aware approximate posterior $r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$.

Variational Inference Maximizing the ELBO is effective as long as the inference gap $D_{\mathrm{KL}}(r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q}) \mid\mid p_\theta(\mathbf{d} \mid \mathbf{a}, \mathbf{q})) = \log p_\theta(\mathbf{a}) - \mathcal{L}_{\mathrm{VI}}(\mathbf{a}, \mathbf{d})$ remains sufficiently small (see Cremer et al. (2018)). Too large of a gap results in a discrepancy between training under $r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$ and evaluating under $p_\theta(\mathbf{d} \mid \mathbf{q})$. Hindsight solves this issue by progressively replacing samples from $r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$ with samples from $p_\theta(\mathbf{d} \mid \mathbf{q})$ during training. Using only samples from $p_\theta(\mathbf{d} \mid \mathbf{q})$ corresponds to using the top-K marginal likelihood. VOD utilizes a tight log-likelihood lower bound (Burda et al., 2015) that we approximate with $\hat{L}_{\alpha=0}^{\mathbb{S}_\phi}$. In VOD, the tight lower bound can be interpolated with a looser bound, the ELBO, by adjusting the parameter α. In contrast to Hindsight, our approach does not require altering the sampling process. Aiding optimization with interpolating the RVB bound has been explored in Liévin et al. (2020).

Indexing, caching and sampling Re-indexing the corpus at every step is prohibitively expensive. Therefore, evaluating the marginal likelihood using samples from the trained retriever is too costly. ORQA, RAG, and RETRO overcome this issue by freezing part of the retriever. REALM, EMDR^2, and Hindsight evaluate the objective asynchronously or periodically cached retriever scores, which results in unaccounted approximations. VOD allows caching the retriever scores while keeping the retriever fully trainable, thanks to modelling the retrieval process explicitly. Last, VOD introduces a truncated retriever parameterization that generalizes the top-K common retrieval approximation and allows handling a retriever using the entire corpus as support. However, we found that, in practice, it is beneficial to restrict retrieval to the top-P documents with $P \geq K$ (exploration/exploitation trade-off).

Table 2: Summary of the medical question answering datasets and the information retrieval benchmark. The number of tokens is measured for the base BioLinkBERT tokenizer, the *mean [min. – max.]* values are reported.

	MedMCQA	**USMLE**	**FZ Queries**
Answer type	multiple-choice	multiple-choice	CUI (UMLS)
# questions (train/valid./test)	182.8k / 4.2k / 6.1k	10.2k / 1.3k / 1.3k	– / – / 248
# tokens per question	19 [4 – 349]	158 [13 – 801]	17 [8 – 89]
# tokens per answer option	5 [3 – 60]	6.6 [3 – 61]	–
Source	AIIMS and NEET PG entrance exams	National Medical Board Examination (US)	search queries registered on FindZebra

5 EXPERIMENTS

We present the tasks and datasets in the medical domain, showcase results on end-to-end multiple-choice ODQA, and apply the trained models to information retrieval. The code for all experiments is available at `http://github.com/vlievin/fz-openqa`.

5.1 DATA

The medical multiple-choice question answering datasets and the corpora are summarized in Table 2.

MedMCQA Pal et al. (2022) is a large-scale multiple-choice question answering collected from Indian medical school entrance exams (AIIMS and NEET-PG). The MedMCQA covers a broad range of medical topics (dentistry, pathology, surgery, preventive medicine, etc.) and many question types (diagnosis, recalling expert factual knowledge, mathematical problem, etc.)

MedWiki We introduce the MedWiki corpus, a collection of 4.5% of articles taken from the English Wikipedia and targeted to the MedMCQA and USMLE datasets. The MedWiki corpus was built by querying each answer option from the MedMCQA and USMLE datasets against the Wikipedia API. Read more in Appendix G.

USMLE Jin et al. (2021)) is a collection of medical questions gathered from the US medical board exam. The questions aim to assess human doctors' medical knowledge and demonstrate decision-making. Each question includes a medical history followed by the vital signs (e.g., blood pressure, temperature), and possibly a specific analysis (e.g., CT-scan).

	MedWiki	**FindZebra**
# articles	293.6k	30.7k
# passages	7,766.9k	711.9k

Table 3: Summary of the medical corpora.

FindZebra corpus & queries FindZebra is a search tool for helping the diagnosis of rare diseases. It is built on open source information retrieval software (BM25) tailored to the problem.[4] The FindZebra corpus indexes a collection of curated articles gathered from GARD, GeneReviews, Genetics Home Reference, OMIM, Orphanet, and Wikipedia. Each article is referenced with a Concept Unique Identifier (CUI)[5]. Additionally, we release a collection of 280 search queries (FZ queries) recorded on the platform. Each query is annotated with a CUI corresponding to the reference search result.

The FindZebra and MedWiki datasets are available at `http://huggingface.co/findzebra`.

5.2 ODQA EXPERIMENTAL SETUP

We implement the retriever using a DPR-like dual-encoder with a shared backbone and implement the multiple-choice reader following Devlin et al. (2018). We use the domain-specific BioLinkBERT Yasunaga et al. (2022) as the backbone for both models and the MedWiki corpus for all QA experiments.

---

[4] `http://findzebra.com`

[5] CUIs are part of the Unified Medical Language System (UMLS, Bodenreider (2004))

All experiments run on a single machine with eight RTX 5000 GPUs using half-precision for training and evaluation. We provide further details in Appendix E.

Hybrid approximate posterior We parameterize the score f_ϕ of the posterior using a composite BM25 score combined with a checkpoint of the retriever score f_θ denoted f_ϕ^{ckpt}. Specifically, we use

$$f_\phi(\mathbf{a}, \mathbf{d}, \mathbf{q}) := f_\phi^{\text{ckpt}}(\mathbf{d}, [\mathbf{q}; \mathbf{a}]) + \tau^{-1} \left(\text{BM25}(\mathbf{q}, \mathbf{d}) + \beta \cdot \text{BM25}(\mathbf{a}, \mathbf{d}) \right) \ . \tag{16}$$

where $\tau = 5$ and β is a parameter scaled proportionally to the ratio of question and answer lengths ${}^{L_\mathbf{q}}/{}_{L_\mathbf{a}}$ to ensure that the BM25 score of the question does not outweigh the answer score. We use $\beta = 1 + 0.5 \max \left\{0, \log \left({}^{L_\mathbf{q}}/{}_{L_\mathbf{a}}\right)\right\}$. At initialization f_θ is uninformative and thus we set $f_\phi^{\text{ckpt}} = 0$

Training, indexing and annealing We organize the training into rounds of T steps as in Khattab et al. (2021). At the beginning of each period, for each question-answer pair $\mathbf{q}_j$, we retrieve the set of top-P documents $\mathbb{T}_\phi$ and cache the set of values $\{f_\phi(\mathbf{d}, \mathbf{q}, \mathbf{a}_j) \mid \mathbf{d} \in \mathbb{T}_\phi\}$, except for the first period where f_ϕ^{ckpt} is set to zero. During the first training round, we anneal the RVB parameter α from 1 to 0 to stabilize early training by distilling the BM25 cached score $f_\phi(\mathbf{a}, \mathbf{d}, \mathbf{q}) = 0 + \tau^{-1} \left(\text{BM25}(\mathbf{q}, \mathbf{d}) + \beta \cdot \text{BM25}(\mathbf{a}, \mathbf{d}) \right)$ into the trainable retriever score $f_\theta(\mathbf{d}, \mathbf{q})$, as pictured in Figure 1. At each training iteration, we sample a set of $K = 8$ document $\mathbb{T}_\phi$ for each of the $M = 4$ question-answer pairs and estimate the RVB and its gradient using the cached values $f_\phi(\mathbf{d}, \mathbf{q}, \mathbf{a}_j)$).

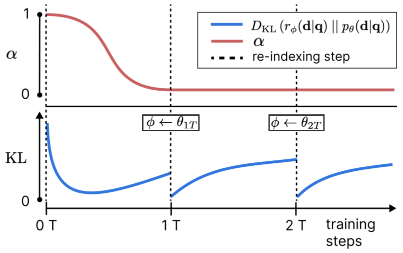


Figure 1: Annealing of the parameter α and typical associated behaviour for the divergence $D_{\text{KL}}(r_\phi(\mathbf{d} \mid \mathbf{q}) || p_\theta(\mathbf{d} \mid \mathbf{q}))$ when ϕ is chosen as a checkpoint of θ, updated every T steps. See empirical data in Appendix D.

Data augmentation The USMLE dataset is small and thus prone to overfitting. We tested training with a concatenated dataset and training first on the MedMCQA, and then on the USMLE.

Baselines We compare the VOD framework with models reported in the literature and with the human baseline. All models reported in the literature are trained disjointly, corresponding to a setup identical to DrQA Chen et al. (2017). We trained disjoint BioLinkBERT readers with a BM25 retrievers by applying VOD with $f_\theta(\mathbf{a}, \mathbf{d}, \mathbf{q}) = f_\phi(\mathbf{a}, \mathbf{d}, \mathbf{q}) = \text{BM25}(\mathbf{q}, \mathbf{d}) + \beta \cdot \text{BM25}(\mathbf{a}, \mathbf{d})$. We report the current state-of-the-art obtained using zero-shot GPT-3 combined with a BM25 retriever and Chain-of-Thought (CoT) prompting Kojima et al. (2022). All the MedMCQA baselines use Wikipedia as a knowledge base, whereas the USMLE baselines use the original MedQA corpus of 18 medical textbooks. We use the MedWiki corpus in all experiments.

Evaluation We estimate the likelihood for each answer option using $C = 10$ Monte-Carlo samples, each containing $MK = 4 \cdot 8 = 32$ documents using the estimates defined in equation 49.

5.3 ODQA ACCURACY

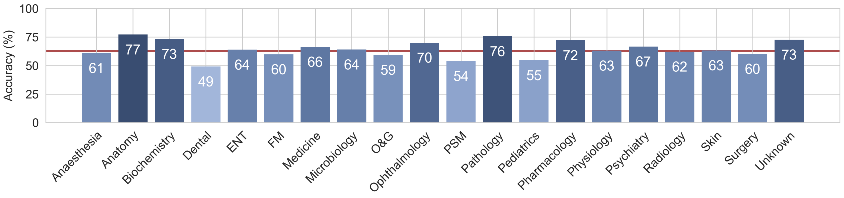


Figure 2: Per-category MedMCQA test accuracy for the VOD model trained on MedMCQA.

Table 4: Open-domain question answering accuracy on the MedMCQA dataset.

Method	Reader	Retriever	QA training set	Valid.	Test
VOD	BioLinkBERT	BM25	MedMCQA	51.6	55.3
VOD	BioLinkBERT	BioLinkBERT	MedMCQA	**58.3**	**62.9**
Uniform baseline	–	–	–	25.0	25.0
Disjoint[1]	PubMedBERT	DPR	MedMCQA	43.0	47.0
Zero-shot prompting[2]	GPT-3	–	–	44.0	–
Zero-shot prompting[2]	GPT-3	BM25	–	46.7	–
Zero-shot CoT prompting[2]	GPT-3	BM25	–	48.8	–
Human (passing score)[2]	–	–	–	≥ 50	≥ 50
Human (merit candidate)[2]	–	–	–	≥ 90	≥ 90

[1]Pal et al. (2022), [2]Liévin et al. (2022)

MedMCQA We report the validation and test accuracy of the VOD framework applied to BioLinkBERT (base) and the baselines in Table 4. VOD outperforms both the disjoint BERT-based and the GPT-3 based with a new state-of-the-art test accuracy of 62.9%, an improvement of +15.9% over the disjoint PubMedBERT reader coupled with a DPR retriever, and +7.6% improvement over the BioLinkBERT reader with static BM25 retriever. Compared to the zero-shot GPT-3 coupled with a simple BM25 retriever, VOD is 9.5% more accurate on the validation set.

In figure 2, we report the accuracy of the VOD model trained on MedMCQA for each of the question categories reported in the test set. VOD performed exceptionally well (>70% test accuracy) for the question related to anatomy, biochemistry, pathology, and pharmacology but performed significantly worse (<60%) in dentistry, paediatrics, obstetrics and gynaecology (O&G), and Preventive & Social Medicine (PSM).

Table 5: Open-domain question answering accuracy on the USMLE dataset.

Method	Reader	Retriever	QA training set	Valid.	Test
VOD	BioLinkBERT	BM25	USMLE	41.0	40.4
VOD	BioLinkBERT	BioLinkBERT	USMLE	45.8	44.7
VOD	BioLinkBERT	BioLinkBERT	MedMCQA	47.2	46.8
VOD	BioLinkBERT	BioLinkBERT	MedMCQA → USMLE	**53.6**	**55.0**
Uniform baseline	–	–	–	25.0	25.0
Custom BM25[1]	–	BM25	–	38.3	36.1
Disjoint[2]	PubMedBERT	BM25	USMLE	–	38.1
Disjoint[3]	BioLinkBERT	BM25	USMLE	–	40.0
Disjoint[3]	BioLinkBERT-L	BM25	USMLE	–	44.6
Zero-shot prompting[4]	GPT-3	–	–	–	46.0
Zero-shot prompting[4]	GPT-3	BM25	–	–	47.3
Zero-shot CoT prompting[4]	GPT-3	BM25	–	–	53.1
Human (passing score)[4]	–	–	–	≥ 60	≥ 60

[1]Jin et al. (2021), [2]Gu et al. (2021), [3]Yasunaga et al. (2022), [4]Liévin et al. (2022)

USMLE We report the validation and test accuracy in Table 5. We found that using VOD with a BioLinkBERT backbone outperforms a BioLinkBERT reader coupled with a BM25 retriever, even when using the larger version of BioLinkBERT (44.7% for VOD, 40.0% for disjoint BioLinkBERT, 44.6% for the disjoint large BioLinkBERT).

We found that pretraining ODQA models on the large MedMCQA results in higher accuracy on the USMLE dataset. A VOD model pretrained on the MedMCQA delivers 46.8% test accuracy in a zero-shot setting and a state-of-the-art of 55.0% test accuracy with further USMLE fine-tuning. This outperforms the zero-shot CoT prompted and retrieval-augmented GPT-3 by +1.9%.

5.4 INFORMATION RETRIEVAL

We tested whether retrievers trained using VOD could be applied to information retrieval. We benchmark deep retrievers against the FindZebra API[6] based on the set of FindZebra queries and corpus. We applied the VOD retriever trained to use question-answer MedMCQA pairs $[\mathbf{q};\mathbf{a}]$ as input and tested an additional model trained using distillation to use queries $\mathbf{q}$ as input. We also evaluated a hybrid retriever with score $f_\theta^{\text{VOD+BM25}}(\mathbf{d},\mathbf{q}) := f_\theta(\mathbf{q},\mathbf{d}) + \tau^{-1}\,\text{BM25}(\mathbf{q},\mathbf{d})$ where $\tau = 5$.

Task-specific distillation We use the VOD retriever trained using question-answer pairs as a teacher to train an additional student BioLinkBERT model using questions only. This corresponds to applying knowledge distillation with the loss:

$$L_{\text{Distill.}} = D_{\text{KL}}(r_\phi(\mathbf{d} \mid [\mathbf{q};\mathbf{a}_\star]) \,||\, p_\theta(\mathbf{d} \mid \mathbf{q}))\ .$$

Metrics We reduced the retrieved passages by article identifier and recorded the rank of the first article linked to a CUI (disease concept) that matches the labels. We report the mean reciprocal rank (MRR) and the fraction of queries for which the correct article is returned in the top 20.

Table 6: Retrieval performances on the FindZebra benchmark for a BioLinkBERT retriever trained using VOD on MedMCQA and one trained using task-specific distillation, with and without coupling with a BM25 score during evaluation.

Method	Distillation	MRR	Hit@20
VOD	✗	27.8	56.9
VOD	✓	31.7	58.1
VOD + BM25	✓	**38.9**	**64.1**
BM25	–	26.4	48.4
FindZebra API	–	30.1	59.3

Performances The FindZebra API implements an advanced custom BM25 score (Dragusin et al., 2013) which powers a specialized search engine utilized by many medical professionals. Nonetheless, a retriever trained using VOD on the MedMCQA dataset is competitive with the API (31.8 MRR for VOD, 30.1 MRR for the API). Coupling the distilled retriever with a simple BM25 baseline gave the best performance with an MRR of 38.9. We found that task-specific distillation was beneficial when applying multiple-choice ODQA retrievers to information retrieval: the task-specific distilled retriever scored 3.9 MRR points above the reference retriever.

Retriever samples In Table 7, we report retrieved top-1 passages for the distilled retriever (two successes and two failures). We found that search results were overall relevant and that terms from the input medical description were utilized and matched beyond simple keyword searches. Nonetheless, the deep retriever often fails when queried with longer comma-separated lists of keywords, as shown in row #4. We speculate that the gap between the training and inference tasks remains large.

6 CONCLUSION

We have introduced VOD, a probabilistic framework for end-to-end training of retrieval augmented models. VOD models the retrieval process explicitly using a target-aware posterior, allowing tractable and consistent estimation of a marginal log-likelihood lower bound. We applied VOD to end-to-end training of multiple-choice ODQA models and scored state-of-the-art results on two datasets. Furthermore, we introduced a medical information retrieval benchmark, which we used to showcase that retrievers trained with VOD can be competitive with industrial search engines.

Although we only applied the framework to the multiple-choice setting with a static approximate posterior, VOD can be applied to a wide range of retrieval-augmented models. Alternatively, the approximate posterior can be learned jointly. We hope our view of retrieval-augmented modelling will spark further interest in designing and training likelihood-based models.

ACKNOWLEDGMENTS

OW's work was funded in part by the Novo Nordisk Foundation through the Center for Basic Machine Learning Research in Life Science (NNF20OC0062606). OW acknowledges support from the Pioneer Centre for AI, DNRF grant number P1.

---

[6] `https://www.findzebra.com/api/`

Table 7: Top-1 passages retrieved for a selection of FindZebra queries with their annotated answer CUIs and the rank of the first matching article for VOD and the FindZebra API. We showcase the retriever model trained with task-specific distillation and without BM25 coupling (MRR 31.7). We highlight terms from the queries and passages relevant to each other.

#	Query	Passage
1	Q: **widespread musculoskeletal pain for more than 6 months and point tenderness in at least 11 of 18 defined anatomical sites** A: Fibromyalgia (C0016053) *Hit rank:* VOD_{BERT}=*1*, FZ_{API}=*1*	*Fibromyalgia.* (...) for IL-1 receptor antagonist, IL-6 and IL-8. Diagnosis The location of the nine paired tender points that comprise the 1990 American College of Rheumatology criteria for fibromyalgia There is no single pathological feature, laboratory finding or biomarker that can diagnose fibromyalgia and there is debate over what should be considered diagnostic criteria and whether an objective diagnosis is possible. In most cases, people with fibromyalgia symptoms may have laboratory test results that appear normal and many of their symptoms may mimic those of other rheumatic conditions such as arthritis or osteoporosis. The most widely accepted set of classification criteria for research purposes was elaborated in 1990 by the Multicenter Criteria Committee of the American College of Rheumatology. These criteria, which are known informally as "the ACR 1990", define fibromyalgia according to the presence of the following criteria: **A history of widespread pain lasting more than three months – affecting all four quadrants of the body, i.e., both sides, and above and below the waist**. Tender points – there (...)
2	Q: diagnosis for dementing syndrome characterized primarily by **impairment of interpersonal and executive function** A: Frontotemporal dementia (C0338451) *Hit rank:* VOD_{BERT}=*1*, FZ_{API}=*8*	*Frontotemporal dementia.* (FTDs) are a group of neurodegenerative disorders associated with shrinking of the frontal and temporal anterior lobes of the brain. **Symptoms include marked changes in social behavior and personality, and/or problems with language**. People with behavior changes may have disinhibition (with socially inappropriate behavior), apathy and loss of empathy, hyperorality (eating excessive amounts of food or attempting to consume inedible things), agitation, compulsive behavior, and various other changes. Examples of problems with language include difficulty speaking or understanding speech. **Some people with FTD also develop a motor syndrome** such as parkinsonism or motor neuron disease (which may be associated with various additional symptoms). There is a strong genetic component to FTDs. It sometimes follows an autosomal dominant inheritance pattern, or sometimes there is a general family history of dementia or psychiatric disorders. The three main genes responsible for familial FTD are MAPT, GRN, and C9orf72. However, the (...)
3	Q: syndrome characterized by **cough, reversible wheezing, and peripheral blood eosinophilia** A: Asthma (C0004096), Reactive airway disease (C3714497) *Hit rank:* VOD_{BERT}=*72*, FZ_{API}=*11*	*Löffler's syndrome.* (...) a parasitic infection such as irritable bowel syndrome, abdominal pain and cramping, skin rashes and fatigue. Löffler's syndrome itself will cause difficulty breathing, coughing as well as a fever. Contents 1 Diagnosis 2 Prevention 3 Epidemiology 4 History 5 See also 6 References 7 External links Diagnosis The diagnosis of Loffler's syndrome can be challenging, as the diagnostic criteria can be vague and consistent with a multitude of diseases or conditions. The disease's developmental trajectory is mostly unknown. Upon examination of symptoms, a doctor will likely request a chest x-ray looking for migratory pulmonary infiltrate, and blood testing, to confirm a diagnosis. Symptoms tend to be brief, but can range from mild to severe and include: fever, vomiting, increased **respirations or difficulty breathing, cough, wheeze**, and rash. Symptoms typically follow an exposure to allergens or certain drugs, and last approximately two weeks. **Eosinophilia is the main feature of diagnostic** (...)
4	Q: 5 year old, boy, congenital malformations, **malformations of the hands** and feet, bilateral strabismus, small tongue, impaired coordination, expressionless face, **prominent forehead**, depressed nasal bridge, hypoplastic thumbs, bilateral adactyly of the feet, **short stature**, severe myopia A: Mobius Syndrome (C0221060), Mobius II syndrome (C0853240) *Hit rank:* $\text{VOD}_{\text{BERT}}=\infty$, FZ_{API}=*1*	*Achondroplasia.* (...) hypochondroplasia, but the features of achondroplasia tend to be more severe. **All people with achondroplasia have short stature**. The average height of an adult male with achondroplasia is 131 centimeters (4 feet, 4 inches), and the average height for adult females is 124 centimeters (4 feet, 1 inch). Characteristic features of achondroplasia include an average-size trunk, short arms and legs with particularly short upper arms and thighs, limited range of motion at the elbows, and an enlarged head (macrocephaly) with a **prominent forehead**. **Fingers are typically short and the ring finger and middle finger may diverge, giving the hand a three-pronged (trident) appearance**. People with achondroplasia are generally of normal intelligence. Health problems commonly associated with achondroplasia include episodes in which breathing slows or stops for short periods (apnea), obesity, (...)

REFERENCES

Olivier Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic acids research*, 32(Database issue):D267–70, January 2004. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkh061.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. December 2021.

T Brown, B Mann, N Ryder, and others. Language models are few-shot learners. *Advances in neural information processing systems*, 2020. ISSN 1049-5258.

Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. September 2015.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer Open-Domain questions. March 2017.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. PaLM: Scaling language modeling with pathways. April 2022.

Chris Cremer, Xuechen Li, and David Duvenaud. Inference suboptimality in variational autoencoders. January 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. October 2018.

Radu Dragusin, Paula Petcu, Christina Lioma, Birger Larsen, Henrik L Jørgensen, Ingemar J Cox, Lars Kai Hansen, Peter Ingwersen, and Ole Winther. FindZebra: A search engine for rare diseases. *International journal of medical informatics*, 82(6):528–538, June 2013. ISSN 1386-5056. doi: 10.1016/j.ijmedinf.2013.01.005.

Nick Duffield, Carsten Lund, and Mikkel Thorup. Priority sampling for estimation of arbitrary subset sums. *Journal of the ACM*, 54(6):32–es, December 2007. ISSN 0004-5411. doi: 10.1145/1314690.1314696.

Víctor Elvira and Luca Martino. Advances in importance sampling. February 2021.

Falcon. PyTorch lightning. *GitHub. Note: https://github. com/PyTorchLightning/pytorch-lightning*.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity, 2021.

Will Grathwohl, Dami Choi, Yuhuai Wu, Geoffrey Roeder, and David Duvenaud. Backpropagation through the void: Optimizing control variates for black-box gradient estimation. October 2017.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-Specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1):1–23, October 2021. ISSN 2691-1957. doi: 10.1145/3458754.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model Pre-Training. In Hal Daumé Iii and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3929–3938. PMLR, 2020.

G E Hinton, P Dayan, B J Frey, and R M Neal. The "wake-sleep" algorithm for unsupervised neural networks. *Science*, 268(5214):1158–1161, May 1995. ISSN 0036-8075. doi: 10.1126/science.7761831.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W Rae, Oriol Vinyals, and Laurent Sifre. Training Compute-Optimal large language models, 2022.

Gautier Izacard and Edouard Grave. Distilling knowledge from reader to retriever for question answering. December 2020a.

Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. July 2020b.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. December 2021.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Few-shot learning with retrieval augmented language models. August 2022.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *APPS. Applied Sciences*, 11(14):6421, July 2021. ISSN 1454-5101, 2076-3417. doi: 10.3390/app11146421.

Jeff Johnson, Matthijs Douze, and Herve Jegou. Billion-scale similarity search with GPUs. *IEEE transactions on big data*, 7(3):535–547, July 2021. ISSN 2332-7790, 2372-2096. doi: 10.1109/tbdata.2019.2921572.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, November 1999. ISSN 0885-6125, 1573-0565. doi: 10.1023/A:1007665907178.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-Tau Yih. Dense passage retrieval for Open-Domain question answering. April 2020.

Omar Khattab and Matei Zaharia. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 39–48. Association for Computing Machinery, New York, NY, USA, July 2020. ISBN 9781450380164. doi: 10.1145/3397271.3401075.

Omar Khattab, Christopher Potts, and Matei Zaharia. Relevance-guided supervision for OpenQA with ColBERT. *Transactions of the Association for Computational Linguistics*, 9:929–944, September 2021. ISSN 2307-387X. doi: 10.1162/tacl_a_00405.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are Zero-Shot reasoners. May 2022.

Wouter Kool, Herke Van Hoof, and Max Welling. Stochastic beams and where to find them: The Gumbel-Top-k trick for sampling sequences without replacement. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3499–3508. PMLR, 2019a.

Wouter Kool, Herke van Hoof, and Max Welling. Buy 4 REINFORCE samples, get a baseline for free! March 2019b.

Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, Eduardo González Ponferrada, Huu Nguyen, Jörg Frohberg, Mario Šaško, Quentin Lhoest, Angelina McMillan-Major, Gérard Dupont, Stella Biderman, Anna Rogers, Loubna Ben allal, Francesco De Toni, Giada Pistilli, Olivier Nguyen, Somaieh Nikpoor, Maraim Masoud, Pierre Colombo, Javier de la Rosa, Paulo Villegas, Tristan Thrush, Shayne Longpre, Sebastian Nagel, Leon Weber, Manuel Romero Muñoz, Jian Zhu, Daniel Van Strien, Zaid Alyafeai, Khalid Almubarak, Vu Minh Chien, Itziar Gonzalez-Dios, Aitor Soroa, Kyle Lo, Manan Dey, Pedro Ortiz Suarez, Aaron Gokaslan, Shamik Bose, David Ifeoluwa Adelani, Long Phan, Ian Yu, Suhas Pai, Violette Lepercq, Suzana Ilic, Margaret Mitchell, Sasha Luccioni, and Yacine Jernite. The BigScience corpus a 1.6TB composite multilingual dataset. June 2022.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. Internet-augmented language models through few-shot prompting for open-domain question answering. March 2022.

Tuan Anh Le, Adam R Kosiorek, N Siddharth, Yee Whye Teh, and Frank Wood. Revisiting reweighted Wake-Sleep for models with stochastic control flow. May 2018.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, February 2020. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/btz682.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 6086–6096, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1612.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. BART: Denoising Sequence-to-Sequence pre-training for natural language generation, translation, and comprehension. October 2019.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-Tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented generation for Knowledge-Intensive NLP tasks. In H Larochelle, M Ranzato, R Hadsell, M F Balcan, and H Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474. Curran Associates, Inc., 2020.

Yingzhen Li and Richard E Turner. Rényi divergence variational inference. In D D Lee, M Sugiyama, U V Luxburg, I Guyon, and R Garnett (eds.), *Advances in Neural Information Processing Systems 29*, pp. 1073–1081. Curran Associates, Inc., 2016.

Opher Lieber, Or Sharir, Barak Lenz, and Yoav Shoham. Jurassic-1: Technical details and evaluation. Technical report, AI21 Labs, August 2021.

Valentin Liévin, Andrea Dittadi, Anders Christensen, and Ole Winther. Optimal variance control of the Score-Function gradient estimator for Importance-Weighted bounds. *Advances in neural information processing systems*, 33:16591–16602, 2020. ISSN 1049-5258.

Valentin Liévin, Christoffer Egeberg Hother, and Ole Winther. Can large language models reason about medical questions? July 2022.

Vaden Masrani, Tuan Anh Le, and Frank Wood. The thermodynamic variational objective. June 2019.

Andriy Mnih and Karol Gregor. Neural variational inference and learning in belief networks. January 2014.

Andriy Mnih and Danilo J Rezende. Variational inference for monte carlo objectives. February 2016.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. MedMCQA: A large-scale Multi-Subject Multi-Choice dataset for medical domain question answering. In Gerardo Flores, George H Chen, Tom Pollard, Joyce C Ho, and Tristan Naumann (eds.), *Proceedings of the Conference on Health, Inference, and Learning*, volume 174 of *Proceedings of Machine Learning Research*, pp. 248–260. PMLR, 2022.

Ashwin Paranjape, Omar Khattab, Christopher Potts, Matei Zaharia, and Christopher D Manning. Hindsight: Posterior-guided training of retrievers for improved open-ended generation. October 2021.

Paszke, Gross, Massa, Lerer, and others. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 2019. ISSN 1049-5258.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. RocketQA: An optimized training approach to dense passage retrieval for Open-Domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5835–5847, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.466.

A Radford, K Narasimhan, T Salimans, and I Sutskever. Improving language understanding by generative pre-training. *cs.ubc.ca*, 2018.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew Johnson, Blake Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Ed Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. Scaling language models: Methods, analysis & insights from training gopher. December 2021.

Devendra Sachan, Mostofa Patwary, Mohammad Shoeybi, Neel Kant, Wei Ping, William L Hamilton, and Bryan Catanzaro. End-to-End training of neural retrievers for Open-Domain question answering, 2021a.

Devendra Singh Sachan, Siva Reddy, William Hamilton, Chris Dyer, and Dani Yogatama. End-to-End training of Multi-Document reader and retriever for Open-Domain question answering. *NeurIPS*, 2021b.

Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, Elton Zhang, Rewon Child, Reza Yazdani Aminabadi, Julie Bernauer, Xia Song, Mohammad Shoeybi, Yuxiong He, Michael Houston, Saurabh Tiwary, and Bryan Catanzaro. Using DeepSpeed and megatron to train Megatron-Turing NLG 530b, a Large-Scale generative language model, 2022.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ambrose Slone, Ameet Rahane, Anantharaman S Iyer, Anders Andreassen, Andrea Madotto, Andrea Santilli, Andreas Stuhlmüller, Andrew Dai, Andrew La, Andrew Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun

Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakaş, B Ryan Roberts, Bao Sheng Loe, Barret Zoph, Bartłomiej Bojanowski, Batuhan Özyurt, Behnam Hedayatnia, Behnam Neyshabur, Benjamin Inden, Benno Stein, Berk Ekmekci, Bill Yuchen Lin, Blake Howald, Cameron Diao, Cameron Dour, Catherine Stinson, Cedrick Argueta, César Ferri Ramírez, Chandan Singh, Charles Rathkopf, Chenlin Meng, Chitta Baral, Chiyu Wu, Chris Callison-Burch, Chris Waites, Christian Voigt, Christopher D Manning, Christopher Potts, Cindy Ramirez, Clara E Rivera, Clemencia Siro, Colin Raffel, Courtney Ashcraft, Cristina Garbacea, Damien Sileo, Dan Garrette, Dan Hendrycks, Dan Kilman, Dan Roth, Daniel Freeman, Daniel Khashabi, Daniel Levy, Daniel Moseguí González, Danielle Perszyk, Danny Hernandez, Danqi Chen, Daphne Ippolito, Dar Gilboa, David Dohan, David Drakard, David Jurgens, Debajyoti Datta, Deep Ganguli, Denis Emelin, Denis Kleyko, Deniz Yuret, Derek Chen, Derek Tam, Dieuwke Hupkes, Diganta Misra, Dilyar Buzan, Dimitri Coelho Mollo, Diyi Yang, Dong-Ho Lee, Ekaterina Shutova, Ekin Dogus Cubuk, Elad Segal, Eleanor Hagerman, Elizabeth Barnes, Elizabeth Donoway, Ellie Pavlick, Emanuele Rodola, Emma Lam, Eric Chu, Eric Tang, Erkut Erdem, Ernie Chang, Ethan A Chi, Ethan Dyer, Ethan Jerzak, Ethan Kim, Eunice Engefu Manyasi, Evgenii Zheltonozhskii, Fanyue Xia, Fatemeh Siar, Fernando Martínez-Plumed, Francesca Happé, Francois Chollet, Frieda Rong, Gaurav Mishra, Genta Indra Winata, Gerard de Melo, Germán Kruszewski, Giambattista Parascandolo, Giorgio Mariani, Gloria Wang, Gonzalo Jaimovitch-López, Gregor Betz, Guy Gur-Ari, Hana Galijasevic, Hannah Kim, Hannah Rashkin, Hannaneh Hajishirzi, Harsh Mehta, Hayden Bogar, Henry Shevlin, Hinrich Schütze, Hiromu Yakura, Hongming Zhang, Hugh Mee Wong, Ian Ng, Isaac Noble, Jaap Jumelet, Jack Geissinger, Jackson Kernion, Jacob Hilton, Jaehoon Lee, Jaime Fernández Fisac, James B Simon, James Koppel, James Zheng, James Zou, Jan Kocoń, Jana Thompson, Jared Kaplan, Jarema Radom, Jascha Sohl-Dickstein, Jason Phang, Jason Wei, Jason Yosinski, Jekaterina Novikova, Jelle Bosscher, Jennifer Marsh, Jeremy Kim, Jeroen Taal, Jesse Engel, Jesujoba Alabi, Jiacheng Xu, Jiaming Song, Jillian Tang, Joan Waweru, John Burden, John Miller, John U Balis, Jonathan Berant, Jörg Frohberg, Jos Rozen, Jose Hernandez-Orallo, Joseph Boudeman, Joseph Jones, Joshua B Tenenbaum, Joshua S Rule, Joyce Chua, Kamil Kanclerz, Karen Livescu, Karl Krauth, Karthik Gopalakrishnan, Katerina Ignatyeva, Katja Markert, Kaustubh D Dhole, Kevin Gimpel, Kevin Omondi, Kory Mathewson, Kristen Chiafullo, Ksenia Shkaruta, Kumar Shridhar, Kyle McDonell, Kyle Richardson, Laria Reynolds, Leo Gao, Li Zhang, Liam Dugan, Lianhui Qin, Lidia Contreras-Ochando, Louis-Philippe Morency, Luca Moschella, Lucas Lam, Lucy Noble, Ludwig Schmidt, Luheng He, Luis Oliveros Colón, Luke Metz, Lütfi Kerem Şenel, Maarten Bosma, Maarten Sap, Maartje ter Hoeve, Maheen Farooqi, Manaal Faruqui, Mantas Mazeika, Marco Baturan, Marco Marelli, Marco Maru, Maria Jose Ramírez Quintana, Marie Tolkiehn, Mario Giulianelli, Martha Lewis, Martin Potthast, Matthew L Leavitt, Matthias Hagen, Mátyás Schubert, Medina Orduna Baitemirova, Melody Arnaud, Melvin McElrath, Michael A Yee, Michael Cohen, Michael Gu, Michael Ivanitskiy, Michael Starritt, Michael Strube, Michał Swędrowski, Michele Bevilacqua, Michihiro Yasunaga, Mihir Kale, Mike Cain, Mimee Xu, Mirac Suzgun, Mo Tiwari, Mohit Bansal, Moin Aminnaseri, Mor Geva, Mozhdeh Gheini, Mukund Varma T, Nanyun Peng, Nathan Chi, Nayeon Lee, Neta Gur-Ari Krakover, Nicholas Cameron, Nicholas Roberts, Nick Doiron, Nikita Nangia, Niklas Deckers, Niklas Muennighoff, Nitish Shirish Keskar, Niveditha S Iyer, Noah Constant, Noah Fiedel, Nuan Wen, Oliver Zhang, Omar Agha, Omar Elbaghdadi, Omer Levy, Owain Evans, Pablo Antonio Moreno Casares, Parth Doshi, Pascale Fung, Paul Pu Liang, Paul Vicol, Pegah Alipoormolabashi, Peiyuan Liao, Percy Liang, Peter Chang, Peter Eckersley, Phu Mon Htut, Pinyu Hwang, Piotr Miłkowski, Piyush Patil, Pouya Pezeshkpour, Priti Oli, Qiaozhu Mei, Qing Lyu, Qinlang Chen, Rabin Banjade, Rachel Etta Rudolph, Raefer Gabriel, Rahel Habacker, Ramón Risco Delgado, Raphaël Millière, Rhythm Garg, Richard Barnes, Rif A Saurous, Riku Arakawa, Robbe Raymaekers, Robert Frank, Rohan Sikand, Roman Novak, Roman Sitelew, Ronan LeBras, Rosanne Liu, Rowan Jacobs, Rui Zhang, Ruslan Salakhutdinov, Ryan Chi, Ryan Lee, Ryan Stovall, Ryan Teehan, Rylan Yang, Sahib Singh, Saif M Mohammad, Sajant Anand, Sam Dillavou, Sam Shleifer, Sam Wiseman, Samuel Gruetter, Samuel R Bowman, Samuel S Schoenholz, Sanghyun Han, Sanjeev Kwatra, Sarah A Rous, Sarik Ghazarian, Sayan Ghosh, Sean Casey, Sebastian Bischoff, Sebastian Gehrmann, Sebastian Schuster, Sepideh Sadeghi, Shadi Hamdan, Sharon Zhou, Shashank Srivastava, Sherry Shi, Shikhar Singh, Shima Asaadi, Shixiang Shane Gu, Shubh Pachchigar, Shubham Toshniwal, Shyam Upadhyay, Shyamolima, Debnath, Siamak Shakeri, Simon Thormeyer, Simone Melzi, Siva Reddy, Sneha Priscilla Makini, Soo-Hwan Lee, Spencer Torene, Sriharsha Hatwar, Stanislas Dehaene, Stefan Divic, Stefano Ermon, Stella Biderman, Stephanie Lin, Stephen Prasad, Steven T Piantadosi, Stuart M Shieber, Summer Misherghi, Svetlana Kiritchenko, Swaroop Mishra, Tal Linzen, Tal Schuster, Tao Li, Tao Yu, Tariq Ali, Tatsu Hashimoto, Te-Lin Wu, Théo Desbordes, Theodore Rothschild, Thomas Phan, Tianle Wang, Tiberius Nkinyili, Timo Schick, Timofei Kornev, Timothy Telleen-Lawton, Titus Tunduny, Tobias Gerstenberg, Trenton Chang, Trishala Neeraj, Tushar Khot, Tyler Shultz, Uri Shaham, Vedant Misra, Vera Demberg, Victoria Nyamai, Vikas Raunak, Vinay Ramasesh, Vinay Uday Prabhu, Vishakh Padmakumar, Vivek Srikumar, William Fedus, William Saunders, William Zhang, Wout Vossen, Xiang Ren, Xiaoyu Tong, Xinran Zhao, Xinyi Wu, Xudong Shen, Yadollah Yaghoobzadeh, Yair Lakretz, Yangqiu Song, Yasaman Bahri, Yejin Choi, Yichi Yang, Yiding Hao, Yifu Chen, Yonatan Belinkov, Yu Hou, Yufang Hou, Yuntao Bai, Zachary Seid, Zhuoye Zhao, Zijian Wang, Zijie J Wang, Zirui Wang, and Ziyi Wu. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models, 2022.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, Yaguang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, Dehao Chen, Yuanzhong Xu, Zhifeng Chen, Adam Roberts, Maarten Bosma, Vincent Zhao, Yanqi Zhou, Chung-Ching Chang, Igor Krivokon, Will Rusch, Marc Pickett, Pranesh Srinivasan, Laichee Man, Kathleen Meier-Hellstern, Meredith Ringel Morris, Tulsee Doshi, Renelito Delos Santos, Toju Duke, Johnny Soraker, Ben Zevenbergen, Vinodkumar Prabhakaran, Mark Diaz, Ben Hutchinson, Kristen Olson, Alejandra Molina, Erin Hoffman-John, Josh Lee, Lora Aroyo, Ravi Rajakumar, Alena Butryna, Matthew Lamm, Viktoriya Kuzmina, Joe Fenton, Aaron Cohen, Rachel Bernstein, Ray Kurzweil, Blaise Aguera-Arcas, Claire Cui, Marian Croak, Ed Chi, and Quoc Le. LaMDA: Language models for dialog applications. January 2022.

George Tucker, Andriy Mnih, Chris J Maddison, John Lawson, and Jascha Sohl-Dickstein. REBAR: Low-variance, unbiased gradient estimates for discrete latent variable models. In I Guyon, U V Luxburg, S Bengio, H Wallach, R Fergus, S Vishwanathan, and R Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 2627–2636. Curran Associates, Inc., 2017.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. November 2017.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. ISSN 1049-5258.

Tim Vieira. Estimating means in a finite universe. `https://timvieira.github.io/blog/post/2017/07/03/estimating-means-in-a-finite-universe/`, 2017. Accessed: 2022-NA-NA.

Sohee Yang and Minjoon Seo. Is retriever merely an approximator of reader? October 2020.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. LinkBERT: Pretraining language models with document links. March 2022.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: Open pre-trained transformer language models, 2022.

A SAMPLING WITHOUT REPLACEMENT WITH PRIORITY SAMPLING

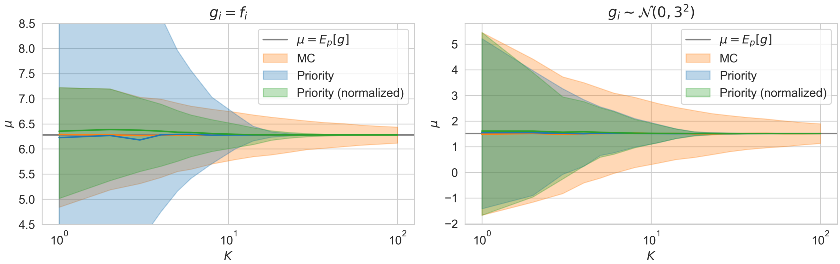


Figure 3: Estimation of $\mu = \mathbb{E}_p[g]$ with $p_i := \sum_{i=1}^{N} {\exp f_i}/{\sum_{j=1} \exp f_j}$, with $f_i \sim \mathcal{N}(0, 3^2)$ and with $N = 100$. We apply standard Monte-Carlo, priority sampling and priority sampling with self-normalized weights. We use $g_i = f_i$ in the left side of the plot, and values $g_i \sim \mathcal{N}(0, 3^2)$ sampled independently of f_i in the right side. We report the 80% CI interval given 10000 estimates, each with $K = 1 \ldots 100$.

Given a set of probabilities $p_1, \ldots, p_N$ and a function with values $f_1, \ldots, f_N$, priority sampling Duffield et al. (2007) allows estimating the sum $\sum_{i=1}^{N} p_i f_i$ using a subset of $K < N$ samples. For a sequence of random weights $u_1, \ldots, u_n \overset{\text{iid}}{\sim} \text{Uniform}(0, 1]$, we define the priority keys p_i/u_i, set τ to be the $K + 1$-th largest key, and define the set of K samples $\mathbb{S} = \{i \in [1, N] \mid p_i/u_i > \tau\}$. Using importance-weights $s_i := \max(p_i, \tau)$, priority sampling is an unbiased estimate as:

$$\mathbb{E}_{p(u_1,\ldots,u_N)}\left[\sum_{i\in\mathbb{S}} s_i f_i\right] = \sum_{i=1}^{N} p_i f_i \,. \tag{17}$$

We recommend Vieira (2017) for a great introduction to priority sampling.

Self-normalized importance sampling Empirically, the estimator 17 might suffer from high variance. We follow Kool et al. (2019a) and use self-normalize importance weights defined as $\tilde{s}_i := {s_i}/{\sum_{j\in\mathbb{S}} s_j}$ to reduce variance at the cost of introducing a bias. However, the estimator $\sum_{i\in\mathbb{S}} \tilde{s}_i f_i$ is biased but consistent: it equals the true expected value for $K = N$. In Figure 3, we visualize the variance of a standard Monte-Carlo (MC) estimator in two cases, a priority sampling estimator and a priority sampling estimator with self-normalized weights. In both cases, the variance of the self-normalized priority estimate is upper-bounded by the variance of the standard MC estimate and converges to zero at a faster rate than the traditional MC estimator. In one of the two cases, the un-normalized priority estimator suffers from large variance.

B ESTIMATION OF THE RVB USING IMPORTANCE SAMPLING

We first summarize the properties of the RVB, notably by documenting its relation with the standard importance-weighted bound. As a second step, we derive the RVB estimates along with their propoerties.

B.1 PROPERTIES OF THE RVB ESTIMATE

Relation to the standard importance-weighted bound Without priority sampling, without the approximation of the normalizing constants, and using a set $\mathbf{d}_1, \ldots, \mathbf{d}_K \sim r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$ sampled with replacement, the original importance-weighted estimate

$$L_\alpha^K(\mathbf{d}, \mathbf{q}) := \frac{1}{1-\alpha} \log \sum_{j=1}^{K} w_{\theta,\phi}^{1-\alpha}(\mathbf{a}, \mathbf{d}_j, \mathbf{q}) \tag{18}$$

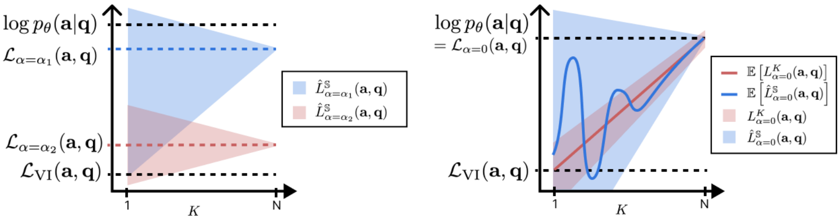


Figure 4: Log-likelihood, ELBO and RVB estimates for $K = |\mathbb{S}|$ samples. **(Left)** RVB for $0 < \alpha_1 < \alpha_2 < 1$ and the corresponding range of values for the importance-weighted estimate $\hat{L}^{\mathbb{S}}_{\alpha=\alpha_1}$ and its estimate $\hat{L}^{\mathbb{S}}_{\alpha=\alpha_2}$. **(Right)** Approximation of the log-likelihood using two RVB estimates: $L^K_{\alpha=0}$ the standard importance-weighted bound (Burda et al., 2015), and our tractable estimation $L^{\mathbb{S}}_{\alpha=0}$. We schematize the range of values for the estimates and their expected values.

of the RVB is a lower-bound of the log-likelihood and for $\alpha = 0$, increasing the number of samples results in a tighter log-likelihood bound (Burda et al., 2015):

$$\log p_\theta(\mathbf{a},\mathbf{q}) \geq L^{K+1}_{\alpha=0}(\mathbf{d},\mathbf{q}) \geq L^{K}_{\alpha=0} \geq \mathcal{L}_{\text{VI}}(\mathbf{a},\mathbf{q})\ . \tag{19}$$

However, the RVB estimate $\hat{L}^{\mathbb{S}}_{\alpha}$ is only an approximation of the original importance-weighted bound.

Consistent estimation of the importance-weighted bound Our approximation to the importance-weighted bound is biased because it involves two approximations:

1. we use priority sampling, which is unbiased but using self-normalized priority sampling is biased but consistent (see section A)
2. we approximate $w_{\theta,\phi}(\mathbf{a},\mathbf{d},\mathbf{q})$ with a biased self-normalizing estimate, which is biased but also consistent (see section B.2.1)

Because of this loss of unbiasedness, and in contrast to the standard importance-weighted bound, the RVB estimate $\hat{L}^{\mathbb{S}}_{\alpha}$ is not guaranteed to be a log-likelihood lower bound. Nonetheless, the RVB, the gradient and the retriever log-density gradient are all consistent: they converge to their respective expected value in the limit $K = N$ (see sections B.2.1 and B.2.2).

In figure 4, we schematize the relationship between the log-likelihood, the RVB, and the range of importance sampling estimates. Even though we cannot guarantee the unbiasedness of the estimate, self-normalized estimates (such as ours) are widely used in the literature. We refer the reader to Elvira & Martino (2021) for an updated review of importance sampling.

Complexity $\mathcal{O}(K)$ All estimates require one backbone (i.e., BERT) call to encode the query and one call to encode each of the K sampled documents. Neglecting the other operations required to evaluate the RVB, this results in a computational complexity of $\mathcal{O}(1 + K) \approx \mathcal{O}(K)$.

B.2 RATIO OF NORMALIZING CONSTANTS

The ratio of un-normalized retriever densities $\zeta(\mathbf{d}) := \exp f_\theta(\mathbf{d},\mathbf{q}) / \exp f_\phi(\mathbf{a},\mathbf{d},\mathbf{q})$ can be used to express the ratio of normalizing constants with the following equality:

$$\frac{\sum_{\mathbf{d}\in\mathbb{T}_\phi} \exp f_\theta(\mathbf{d},\mathbf{q})}{\sum_{\mathbf{d}'\in\mathbb{T}_\phi} \exp f_\phi(\mathbf{a},\mathbf{d}',\mathbf{q})} = \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[\zeta(\mathbf{d})\right]\ . \tag{20}$$

We use the above equality to estimate $w_{\theta,\phi}(\mathbf{a},\mathbf{d},\mathbf{q})$ and $\nabla_\theta \log p_\theta(\mathbf{d} \mid \mathbf{q})$ during training and evaluation. The equality arises from the definition of the right-hand term:

$$\begin{aligned}
\mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[\zeta(\mathbf{d})\right] &:= \sum_{\mathbf{d}\in\mathbb{T}_\phi} r_\phi(\mathbf{d} \mid \mathbf{a},\mathbf{q}) \frac{\exp f_\theta(\mathbf{d},\mathbf{q})}{\exp f_\phi(\mathbf{a},\mathbf{d},\mathbf{q})} && (21)\\
&= \sum_{\mathbf{d}\in\mathbb{T}_\phi} \frac{\exp f_\phi(\mathbf{a},\mathbf{d},\mathbf{q})}{\sum_{\mathbf{d}'\in\mathbb{T}_\phi} \exp f_\phi(\mathbf{a},\mathbf{d}',\mathbf{q})} \frac{\exp f_\theta(\mathbf{d},\mathbf{q})}{\exp f_\phi(\mathbf{a},\mathbf{d},\mathbf{q})} \, . && (22)
\end{aligned}$$

B.2.1 RVB ESTIMATE

We apply self-normalized priority sampling (see Appendix A) to estimate $\mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[w_{\theta,\phi}^{1-\alpha}(\mathbf{a},\mathbf{d},\mathbf{q})\right]$. We denote $\mathbb{S} = \{\mathbf{d}_1,\ldots,\mathbf{d}_K\}$ the set of documents sampled without replacement and $\tilde{s}(\mathbf{d}_1),\ldots,\tilde{s}(\mathbf{d}_K)$ the self-normalized priority importance weights. This gives an estimate of the RVB that is consistent (i.e., it converges to the true value in the limit $K = N$ with probability 1):

$$\begin{aligned}
\mathcal{L}_\alpha(\mathbf{a},\mathbf{q}) :=& \frac{1}{1-\alpha} \log \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[w_{\theta,\phi}^{1-\alpha}(\mathbf{a},\mathbf{d},\mathbf{q})\right] && (23)\\
\approx& \frac{1}{1-\alpha} \log \sum_{\mathbf{d}\in\mathbb{S}} \tilde{s}(\mathbf{d}) w_{\theta,\phi}^{1-\alpha}(\mathbf{a},\mathbf{d},\mathbf{q}) := \hat{L}_\alpha^{\mathbb{S}}(\mathbf{a},\mathbf{q}) \, . && (24)
\end{aligned}$$

Evaluating the weights $w_{\theta,\phi}(\mathbf{a},\mathbf{d},\mathbf{q})$ requires estimating the normalizing constant of $p_\theta(\mathbf{d} \mid \mathbf{q})$, which would require P backbone (e.g. BERT) calls. Using the identity 20, we obtain a self-normalized estimate of the weight:

$$\begin{aligned}
w_{\theta,\phi}(\mathbf{a},\mathbf{d},\mathbf{q}) :=& \frac{p_\theta(\mathbf{a} \mid \mathbf{d},\mathbf{q}) p_\theta(\mathbf{d} \mid \mathbf{q})}{r_\phi(\mathbf{d} \mid \mathbf{a},\mathbf{q})} && (25)\\
=& p_\theta(\mathbf{a} \mid \mathbf{d},\mathbf{q}) \zeta(\mathbf{d}) \left(\frac{\sum_{\mathbf{d}'\in\mathbb{T}_\phi} \exp f_\theta(\mathbf{d}',\mathbf{q})}{\sum_{\mathbf{d}'\in\mathbb{T}_\phi} \exp f_\phi(\mathbf{a},\mathbf{d},\mathbf{q}')} \right)^{-1} && (26)\\
=& p_\theta(\mathbf{a} \mid \mathbf{d},\mathbf{q}) \zeta(\mathbf{d}) \left(\mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[\zeta(\mathbf{d})\right] \right)^{-1} && (27)\\
\approx& \frac{p_\theta(\mathbf{a} \mid \mathbf{d},\mathbf{q}) \zeta(\mathbf{d})}{\sum_{\mathbf{d}'\in\mathbb{T}_\phi} \tilde{s}(\mathbf{d}') \zeta(\mathbf{d}')} \, . && (28)
\end{aligned}$$

This estimate requires only $K \leq P$ backbone calls and is consistent because the denominator $\mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[\zeta(\mathbf{d})\right]$ is estimated using a consistent self-normalized priority sampling estimator. Finally, by combining the two previous steps, we obtain the estimate of the RVB defined in Equation 10. The resulting RVB estimate is consistent because it combines consistent estimators.

B.2.2 RVB GRADIENT ESTIMATE

Using the results from the previous section, the gradient of the RVB w.r.t the parameter θ can be estimated as:

$$\begin{aligned}
\nabla_\theta \mathcal{L}_\alpha(\mathbf{a},\mathbf{q}) :=& \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[\widetilde{w_{\theta,\phi}^{1-\alpha}}(\mathbf{a},\mathbf{d},\mathbf{q}) \, \nabla_\theta \log p_\theta(\mathbf{a},\mathbf{d} \mid \mathbf{q})\right] && (29)\\
:=& \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[\frac{w_{\theta,\phi}^{1-\alpha}(\mathbf{a},\mathbf{d},\mathbf{q})}{\mathbb{E}_{r_\phi(\mathbf{d}'|\mathbf{a},\mathbf{q})}\left[w_{\theta,\phi}^{1-\alpha}(\mathbf{a},\mathbf{d}',\mathbf{q})\right]} \, \nabla_\theta \log p_\theta(\mathbf{a},\mathbf{d} \mid \mathbf{q})\right] && (30)\\
\approx& \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})}\left[\frac{\left({}^{p_\theta(\mathbf{a}|\mathbf{d},\mathbf{q})\zeta(\mathbf{d})}/{}_{\sum_{\mathbf{d}''\in\mathbb{T}_\phi} \tilde{s}(\mathbf{d}'')\zeta(\mathbf{d}')}\right)^{1-\alpha}}{\mathbb{E}_{r_\phi(\mathbf{d}'|\mathbf{a},\mathbf{q})}\left[\left({}^{p_\theta(\mathbf{a}|\mathbf{d}',\mathbf{q})\zeta(\mathbf{d}')}/{}_{\sum_{\mathbf{d}''\in\mathbb{T}_\phi} \tilde{s}(\mathbf{d}'')\zeta(\mathbf{d}')}\right)^{1-\alpha}\right]} \, \nabla_\theta \log p_\theta(\mathbf{a},\mathbf{d} \mid \mathbf{q})\right] && (31)\\
\approx& \sum_{\mathbf{d}\in\mathbb{S}} \frac{\tilde{s}(\mathbf{d}) \left(p_\theta(\mathbf{a} \mid \mathbf{d},\mathbf{q})\zeta(\mathbf{d})\right)^{1-\alpha}}{\sum_{\mathbf{d}'\in\mathbb{S}} \tilde{s}(\mathbf{d}) \left(p_\theta(\mathbf{a} \mid \mathbf{d}',\mathbf{q})\zeta(\mathbf{d})\right)^{1-\alpha}} \nabla_\theta \log p_\theta(\mathbf{a},\mathbf{d} \mid \mathbf{q}) \, . && (32)
\end{aligned}$$

Another approximation is required to estimate $\log p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q}) = \log p_\theta(\mathbf{q} \mid \mathbf{d}, \mathbf{q}) + \log p_\theta(\mathbf{d} \mid \mathbf{q})$ without evaluating $\sum_{\mathbf{d} \in \mathbb{T}_\phi} \exp f_\theta(\mathbf{d}, \mathbf{q})$. This approximation is also consistent because self-normalized priority sampling is consistent:

$$\begin{aligned}
\nabla_\theta \log p_\theta(\mathbf{d} \mid \mathbf{q}) =& \nabla_\theta f_\theta(\mathbf{d}, \mathbf{q}) - \nabla_\theta \log \sum_{\mathbf{d}' \in \mathbb{T}_\phi} \exp f_\theta(\mathbf{d}', \mathbf{q}) && (33) \\
=& \nabla_\theta f_\theta(\mathbf{d}, \mathbf{q}) - \frac{\nabla_\theta \sum_{\mathbf{d}' \in \mathbb{T}_\phi} \exp f_\theta(\mathbf{d}, \mathbf{q})}{\sum_{\mathbf{d}'' \in \mathbb{T}_\phi} \exp f_\theta(\mathbf{d}'', \mathbf{q})} && (34) \\
=& \nabla_\theta f_\theta(\mathbf{d}, \mathbf{q}) - \sum_{\mathbf{d}' \in \mathbb{T}_\phi} p_\theta(\mathbf{d}' \mid \mathbf{q}) \nabla_\theta f_\theta(\mathbf{d}', \mathbf{q}) && (35) \\
=& \nabla_\theta f_\theta(\mathbf{d}, \mathbf{q}) - \sum_{\mathbf{d}' \in \mathbb{T}_\phi} r_\phi(\mathbf{d}' \mid \mathbf{a}, \mathbf{q}) \frac{p_\theta(\mathbf{d}' \mid \mathbf{q})}{r_\phi(\mathbf{d}' \mid \mathbf{a}, \mathbf{q})} \nabla_\theta f_\theta(\mathbf{d}', \mathbf{q}) && (36) \\
=& \nabla_\theta f_\theta(\mathbf{d}, \mathbf{q}) - \sum_{\mathbf{d}' \in \mathbb{T}_\phi} r_\phi(\mathbf{d}' \mid \mathbf{a}, \mathbf{q}) \frac{\zeta(\mathbf{d}')}{\mathbb{E}_{r_\phi(\mathbf{d}'' \mid \mathbf{a}, \mathbf{q})}\left[\zeta(\mathbf{d}'')\right]} \nabla_\theta f_\theta(\mathbf{d}', \mathbf{q}) && (37) \\
\approx& \nabla_\theta f_\theta(\mathbf{d}, \mathbf{q}) - \sum_{\mathbf{d}' \in \mathbb{T}_\phi} \frac{\tilde{s}(\mathbf{d}')\zeta(\mathbf{d}')}{\sum_{\mathbf{d}'' \in \mathbb{T}_\phi} \tilde{s}(\mathbf{d}'')\zeta(\mathbf{d}'')} \nabla_\theta f_\theta(\mathbf{d}', \mathbf{q}) \, . && (38)
\end{aligned}$$

C APPLICATIONS OF THE VOD FRAMEWORK

In this section, we detail how to apply the VOD framework to the tasks of language modelling as well as extractive, generative and multiple-choice ODQA. We also detail a solution to optimizing multi-documents readers (FiD) jointly.

C.1 GENERATIVE AND EXTRACTIVE ODQA

The model $p_\theta(\mathbf{a}|\mathbf{d}, \mathbf{q})$ a machine reading comprehension component that can be implemented either using an extractive approach, as done in the original BERT (Devlin et al., 2018), or using a generative approach Lewis et al. (2019). Applying the VOD framework to generative and extractive ODQA simply requires plugging the likelihood of the corresponding machine reading comprehension model $p_\theta(\mathbf{a}|\mathbf{d}, \mathbf{q})$ in the RVB and gradient estimates (equations 10 and 11).

C.2 RETRIEVAL-AUGMENTED LANGUAGE MODELLING

We consider the variable $\mathbf{a} = [\mathbf{a}_1, \ldots, \mathbf{a}_T]$ to be the sequence of tokens of length T and omit the condition $\mathbf{q}$. We consider a left-to-right factorization $p_\theta(\mathbf{a}) := \prod_{t=1}^{T} p_\theta(\mathbf{a}_t|\mathbf{a}_{<t})$ and define the following retrieval-augmented language model with one retrieved document per token:

$$p_\theta(\mathbf{a}) := \prod_{t=1}^{T} \sum_{\mathbf{d}_t \in \mathbb{D}} p_\theta(\mathbf{d}_t|\mathbf{a}_{<t}) p_\theta(\mathbf{a}_t|\mathbf{d}_t, \mathbf{a}_{<t}) \, . \tag{39}$$

We apply the RVB to each step t using an approximate posterior $r_\phi(\mathbf{d}_t|\mathbf{a})$, this results in the following log-likelihood lower bound for $\alpha \geq 0$:

$$\begin{aligned}
\log p_\theta(\mathbf{a}) \geq& \log \prod_{t=1}^{T} \mathcal{L}_\alpha(\mathbf{a}_t, \mathbf{a}_{<t}) && (40) \\
=& \frac{1}{1-\alpha} \sum_{t=1}^{T} \log \mathbb{E}_{r_\phi(\mathbf{d}|\mathbf{a},\mathbf{q})} \left[w_{\theta,\phi}^{1-\alpha}(\mathbf{a}_t, \mathbf{d}_t, \mathbf{a}_{<t}) \right] \, . && (41)
\end{aligned}$$

The above step-wise RVB $\mathcal{L}_\alpha(\mathbf{a}_t, \mathbf{a}_{<t})$ can be estimated using equations 10 and 11.

C.3 MULTIPLE-CHOICE ODQA

In the multiple-choice setting, a vector of M answer options $\mathbf{A} := [\mathbf{a}_1, \ldots, \mathbf{a}_M]$ is given. We denote $\mathbf{a}$ the correct option and assume $\mathbf{a} \in \mathbf{A}$. We define the vector of M queries as $\mathbf{Q} = [\mathbf{q}_1, \ldots, \mathbf{q}_M]$ with $\mathbf{q}_j := [\mathbf{q}; \mathbf{a}_j]$ where $[\cdot;\cdot]$ denotes the concatenation operator. We denote $\times$ the Cartesian product, $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_M]$ a vector of M documents, and $\mathbb{D}^{(M)} := \mathbb{D} \times \ldots \times \mathbb{D}$ the set of combinations of vectors of M documents (N^M document vectors). We choose:

$$\begin{aligned}
p_\theta(\mathbf{a}_\star | \mathbf{D}, \mathbf{Q}) :=& \frac{\exp g_\theta(\mathbf{d}_\star, \mathbf{q}_\star)}{\sum_{j=1}^M \exp g_\theta(\mathbf{d}_j, \mathbf{q}_j)} && (42)\\
p_\theta(\mathbf{D} | \mathbf{Q}) :=& \prod_{j=1}^M p_\theta(\mathbf{d}_j | \mathbf{q}_j) && (43)\\
r_\phi(\mathbf{D} | \mathbf{A}, \mathbf{Q}) :=& \prod_{j=1}^M r_\phi(\mathbf{d}_j | \mathbf{q}_j)\,. && (44)
\end{aligned}$$

Given a set of K documents $\mathbb{S}_j$ sampled without replacement from $r_\phi(\mathbf{d}_j|\mathbf{q}_j)$ for each option j and with priority weights $s_j(\mathbf{d}_j)$, we denote $\mathbb{S}^{(M)} := \mathbb{S}_1 \times \ldots \times \mathbb{S}_M$ the Cartesian product of the sets of per-option samples. Because the components of $\mathbf{D}$ are independent (equation 44), priority sampling can be applied to the product $r_\phi(\mathbf{D}|\mathbf{Q})$ with self-normalized weights $\tilde{s}(\mathbf{D}) := \prod_{j=1}^M s_j(\mathbf{d}_j)$. For an arbitrary function $h(\mathbf{D})$, we have:

$$\begin{aligned}
\mathbb{E}_{r_\phi(\mathbf{D}|\mathbf{Q})}\left[h(\mathbf{D})\right] =& \mathbb{E}_{r_\phi(\mathbf{d}_1|\mathbf{q}_1)}\left[\ldots \mathbb{E}_{r_\phi(\mathbf{d}_M|\mathbf{q}_M}\left[h(\mathbf{D})\right]\ldots\right]\\
\approx& \sum_{\mathbf{d}_1 \in \mathbb{D}} \tilde{s}(\mathbf{d}_1) \ldots \sum_{\mathbf{d}_M \in \mathbb{D}} \tilde{s}(\mathbf{d}_M) h(\mathbf{D})\\
=& \sum_{\mathbf{D} \in \mathbb{S}^{(M)}} \tilde{s}(\mathbf{D}) h(\mathbf{D})\,.
\end{aligned}$$

We have defined a model for multiple-choice and showed that priority sampling can be applied to the product $r_\phi(\mathbf{D}|\mathbf{Q})$ with weights $\tilde{s}(\mathbf{D}) := \prod_{j=1}^M s_j(\mathbf{d}_j)$, we will now show how the RVB estimates can be adapted to the multiple-choice setting. The derivation is identical to the one presented in section B. This arises from the fact that there are functions $F_\theta(\mathbf{D}, \mathbf{Q}) := \sum_{j=1}^M f_\theta(\mathbf{d}_j, \mathbf{q}_j)$ and $F_\phi(\mathbf{A}, \mathbf{D}, \mathbf{Q}) := \sum_{j=1}^M f_\phi(\mathbf{d}_j, \mathbf{q}_j)$ and a set $\boldsymbol{\mathcal{T}}_\phi := \{\mathbf{D} \in \mathbb{D}^{(M)} | \forall j \in [1, M], \mathbf{d} \in \mathbb{T}_\phi(\mathbf{q}_j)\}$, the ensemble of the top-P documents for each query $\mathbf{q}_j$, such that

$$\begin{aligned}
p_\theta(\mathbf{D}|\mathbf{Q}) \propto& \mathbb{1}[\mathbf{D} \in \boldsymbol{\mathcal{T}}_\phi] \exp F_\theta(\mathbf{D}, \mathbf{Q}) && (45)\\
r_\phi(\mathbf{D}|\mathbf{A}, \mathbf{Q}) \propto& \mathbb{1}[\mathbf{D} \in \boldsymbol{\mathcal{T}}_\phi] \exp F_\phi(\mathbf{A}, \mathbf{D}, \mathbf{Q})\,. && (46)
\end{aligned}$$

By applying the results from section B to $\mathbf{A}, \mathbf{D}, \mathbf{Q}$ with $\zeta(\mathbf{D}) = \exp F_\theta(\mathbf{D}, \mathbf{Q}) / \exp F_\phi(\mathbf{A}, \mathbf{D}, \mathbf{Q})$ the RVB and its gradients can be estimated using:

$$\mathcal{L}_\alpha(\mathbf{a}_\star, \mathbf{Q}) \approx \hat{L}_\alpha^{\mathbb{S}^{(M)}}(\mathbf{a}_\star, \mathbf{Q}) := \frac{1}{1-\alpha} \log \sum_{\mathbf{D} \in \mathbb{S}^{(M)}} \tilde{s}(\mathbf{D}) \left(\frac{\zeta(\mathbf{D}) p_\theta(\mathbf{A}|\mathbf{D}, \mathbf{Q})}{\sum_{\mathbf{D}' \in \mathbb{S}^{(M)}} \tilde{s}(\mathbf{D}') \zeta(\mathbf{D}')} \right)^{1-\alpha} \tag{47}$$

$$\nabla_\theta \mathcal{L}_\alpha(\mathbf{A}, \mathbf{Q}) \approx \sum_{\mathbf{D} \in \mathbb{S}^{(M)}} \frac{\tilde{s}(\mathbf{D}) \left(\zeta(\mathbf{D}) p_\theta(\mathbf{A}|\mathbf{D}, \mathbf{Q})\right)^{1-\alpha}}{\sum_{\mathbf{D}' \in \mathbb{S}^{(M)}} \tilde{s}(\mathbf{D}') \left(\zeta(\mathbf{D}') p_\theta(\mathbf{A}|\mathbf{D}', \mathbf{Q})\right)^{1-\alpha}} \nabla_\theta \log p_\theta(\mathbf{A}, \mathbf{D}|\mathbf{Q})\,. \tag{48}$$

Monte-Carlo estimation During training, the computational budget is tight, and the RVB and its gradient are estimated using a single set of samples $\mathbb{S}^{(M)}$. During evaluation, we can leverage $C \geq 1$ Monte-Carlo samples $\mathbb{S}_1^M, \ldots, \mathbb{S}_C^M$, each containing K^M documents sampled from $r_\phi(\mathbf{D}|\mathbf{A}, \mathbf{Q})$ without replacement, to estimate the RVB (and therefore the log-likelihood) more accurately. We use the following estimate:

$$\hat{p}_\theta(\mathbf{a}, \mathbf{Q}) := \frac{1}{C} \sum_{i=1}^C \frac{\exp \hat{L}_\alpha^{\mathbb{S}_i^{(M)}}(\mathbf{a}, \mathbf{q})}{\sum_{\mathbf{a}' \in \mathbf{A}} \exp \hat{L}_\alpha^{\mathbb{S}_i^{(M)}}(\mathbf{a}', \mathbf{Q})}\,. \tag{49}$$

C.4 FUSION-IN-DECODER (FID)

In this work, we considered reader models $p_\theta(\mathbf{a}|\mathbf{d},\mathbf{q})$ with a single document per sample. Alternatively, models such as FiD (Izacard & Grave, 2020b) implement a reader model that allows reading multiple documents per sample. Given a set $\mathbb{S} := \{\mathbf{d}_1, \ldots, \mathbf{d}_K\}$ of documents, we denote the multi-document reader $p_\theta(\mathbf{a}|\mathbb{S},\mathbf{q})$. Defining a distribution over the set of unique documents $p(\mathbb{S})$ with tractable sampling and density evaluation is challenging. EMDR^2 (Sachan et al., 2021b) optimized a multi-document reader jointly with a deep retriever. However, an auxiliary reader model $p_\theta(\mathbf{a}|\mathbb{S},\mathbf{q}) := \prod_{i=1}^{K} p_\theta(\mathbf{a}|\mathbf{d}_i,\mathbf{q})$ is used to optimize a retriever model $p_\theta(\mathbb{S}|\mathbf{q}) := \prod_{i=1}^{K} p_\theta(\mathbf{d}_i|\mathbf{q})$. VOD can be applied by following the same strategy, and this is equivalent to optimizing a single-sample joint reader along with a multi-sample reader:

$$L_{\text{MULTISAMPLE}} := \underbrace{\nabla_\theta \log p_\theta(\mathbf{a}|\mathbb{S},\mathbf{q})}_{\substack{\text{multi-sample}\\ \text{reader likelihood}}} + \underbrace{\sum_{\mathbf{d}\in\mathbb{S}} \tilde{s}(\mathbf{d})\, \widetilde{w^{1-\alpha}_{\theta,\phi}}(\mathbf{a},\mathbf{d}|\mathbb{S})\; \nabla_\theta \log p_\theta(\mathbf{a},\mathbf{d}|\mathbf{q})}_{\text{single-sample RVB gradient}} \; . \tag{50}$$

D EXPERIMENTAL RESULTS

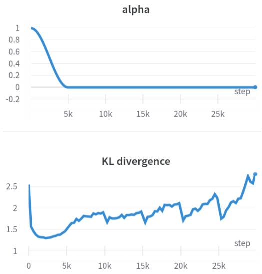


Figure 5: Measure of the divergence $D_{\text{KL}}\left(r_\phi(\mathbf{d}|\mathbf{q}) \,||\, p_\theta(\mathbf{d}|\mathbf{q})\right)$ during the training of a VOD retriever on the USMLE dataset. The retriever checkpoint is updated every $T = 5k$ steps. α is annealed from 1 to 0 during the first 5k steps. We recognize the pattern schematized in Figure 1. In this example, the approximate posterior is chosen as a combination of a checkpoint of the retriever and a static BM25 component. Therefore the value of the divergence is never zero because the divergence between the model and the BM25 retriever is always strictly positive.

E IMPLEMENTATION

Table 8: Parameterization of the reader and retriever scores. The complexity is reported for a batch-size of one, M answer option, and for K documents and inputs $\mathbf{q}_j = [\mathbf{q}; \mathbf{a}_j]$ and $\mathbf{d}$ of lengths $L_\mathbf{q}$ and $L_\mathbf{a}$. When using a dual-encoder architecture, the parameters are shared across the two encoders.

Type	**Complexity**	**Parameterization**
dual-encoder	$M(L_\mathbf{q}^2 + KL_\mathbf{d}^2)$	$f_\theta(\mathbf{d},\mathbf{q}_j) = \text{Linear}_{\theta\|D}(\text{BERT}_\theta(\mathbf{d}))^T \text{Linear}_{\theta\|Q}(\text{BERT}_\theta(\mathbf{q}_j))$
Cross attn.	$MK(L_\mathbf{q} + L_\mathbf{d})^2$	$g_\theta(\mathbf{d},\mathbf{q}_j) = \text{Linear}_\theta(\text{BERT}_\theta([\mathbf{d}; \mathbf{q}_j]))$

Documents preprocessing We encode the text and title of all the articles using the relevant BERT tokenizer. For each article with encoded title $\mathbf{t}$ of length $L_\mathbf{t}$, we extract overlapping passages $\mathbf{p}$ of length $200 - 1 - \mathbf{t}$ with stride 100 tokens. For each passage, using `[DOC]` a special token added to the BERT vocabulary, we format each passage as

$$\mathbf{d} := [\texttt{[CLS]}\ ;\ \texttt{[DOC]}\ ;\ \mathbf{t}\ ;\ \mathbf{p}]\ . \tag{51}$$

Queries preprocessing We encode all questions and answer options using the tokenizer and store the question-answer pairs as

$$\mathbf{q}_j := [\texttt{[CLS]}\ ;\ \texttt{[QUERY]}\ ;\ \mathbf{q}\ ;\ \texttt{[SEP]}\ ;\ \mathbf{a}_j] \tag{52}$$

where the question $\mathbf{q}$ is truncated such as $|\mathbf{q}_j| \leq 312$ tokens and `[QUERY]` is an additional special token. On the reader side, we append the document passage $\mathbf{d}$ to the question-answer query q_j such that $\mathbf{q}_j := [\mathbf{d}; \texttt{[SEP]}; \texttt{[QUERY]}; \mathbf{q}; \texttt{[SEP]}; \mathbf{a}_j]$.

Reader We parameterize the reader score g_θ using a cross-attention model parameterized by another BERT backbone. Each query $\mathbf{q}_j = [\mathbf{q}; \mathbf{a}_j]$ is prepended with a document $\mathbf{d}$, and an additional linear layer is used to reduce the output of BERT at the `CLS` token to a scalar value, as originally done in Devlin et al. (2018). See expression in Table 8.

Retriever We parameterize the retriever score f_θ using a dual encoder architecture similar to DPR, except that we share the BERT backbone across the two columns and one linear layer to project the output of each column. See expression in Table 8.

Hyperparameters We summarize the training, evaluation and model hyperparameters in Table 9.

Table 9: Hyperparameters used across the multiple-choice ODQA experiments.

Category	**Parameter**	**Value**
Optimization	Optimizer	AdamW
	Learning rate	$3 \cdot 10^{-6}$
	Learning rate warmup	$0.1 \cdot T$
	Warmup frequency	every T steps
	Weight decay	$1 \cdot 10^{-3}$
	Gradient clipping	0.5
	Precision	`float16`
α annealing	initial value	1
	final value	0
	length	T steps
	type	cosine
Model	Reader	BioLinkBERT + linear layer
	Retriever	BioLinkBERT + two linear layers
	Output vector size	768
Batching	batch-size	32
	M (# of options)	4
	K (documents per option)	8
	P (retriever support size)	100
	N (corpus size)	7,766.9k
	document passage stride	100
	$L_{\mathbf{d}}$ (document passage length)	200
	max. $L_{\mathbf{q}}$ (max. query length)	312
	max. $L_{\mathbf{d}} + L_{\mathbf{q}}$	512
Training	T (re-indexing period length)	5k
	Training steps (MedMCQA)	150k
	Training steps (USMLE)	50k
	Training steps (MedMCQA →USMLE)	150k → 10k
	Training steps (MedMCQA + USMLE)	–
	Training steps (Distillation)	120k
Posterior and retrieval	parameterization	$f_\phi^{\text{ckpt}}(\mathbf{d}, [\mathbf{q}; \mathbf{a}]) + \tau^{-1}\left(\text{BM25}(\mathbf{q}) + \beta \cdot \text{BM25}(\mathbf{a})\right)$
	τ (BM25 temperature)	5
	β (BM25 answer weight)	$1 + 0.5 \max\left\{0, \log\left({}^{L_{\mathbf{q}}}/_{\mathrm{L}_{\mathbf{a}}}\right)\right\}$
	BM25 implementation	`elasticsearch` v7.14.1
	BM25 paramters	b=0.75, k1=1.2
	MIPS implementation	`faiss v1.7.2`
	`faiss` factory string	IVF1000,Flat
	`faiss` precision	`float16`
	`faiss` nprobe	32
Evaluation	C (Monte-Carlo samples for eval.)	10
Hardware	CPU	AMD EPYC 7252 8-Core Processor
	RAM	256 GB
	GPU	8 × Quadro RTX 5000
	VRAM	128 GB
Software	PyTorch	Paszke et al. (2019)
	Lightning	Falcon
	`faiss`	Johnson et al. (2021)

F SYMBOLS

Table 10: Mathematical symbols.

Category	**Symbol**	**Description**
ODQA variables	$\mathbf{a}$	answer
	$\mathbf{d}$	document or document passage
	$\mathbf{q}$	question or query
	$L_{\mathbf{a}}$	number of tokens in the answer
	$L_{\mathbf{d}}$	number of tokens in the document
	$L_{\mathbf{q}}$	number of tokens in the query
	$\mathbb{D}$	corpus of documents
	N	number of documents in the corpus
Reader-retriever	θ	parameter of the retrieval-augmented model (generative model)
	$p_\theta(\mathbf{a}, \mathbf{d} \mid \mathbf{q})$	Joint reader-retriever model
	$w_{\theta,\phi}(\mathbf{a}, \mathbf{d}, \mathbf{q})$	Importance weight
	$\zeta(\mathbf{d})$	ratio of exponentiated scores $\exp f_\theta(\mathbf{d}, \mathbf{q}) / \exp f_\phi(\mathbf{a}, \mathbf{d}, \mathbf{q})$
	$p_\theta(\mathbf{a} \mid \mathbf{d}, \mathbf{q})$	reader
	$p_\theta(\mathbf{d} \mid \mathbf{q})$	retriever
	$f_\theta(\mathbf{d}, \mathbf{q})$	score of the retriever
Posterior	ϕ	parameter of the approximate posterior (inference network)
	$r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$	approximate posterior (static retriever)
	$f_\phi(\mathbf{a}, \mathbf{d}, \mathbf{q})$	score of the approximate posterior
	$\mathrm{BM25}(\mathbf{q}, \mathbf{d})$	BM25 score of the query $\mathbf{q}$ for the document $\mathbf{d}$
	$f_\phi^{\mathrm{ckpt}}(\mathbf{d}, \mathbf{q})$	checkpoint of the retriever f_θ at step $k \cdot T$
	τ	temperature balancing the checkpoint score and the BM25 score
	β	weight balancing the query and answer options BM25 terms
Truncated retriever	P	number of documents with non-zero mass under $p_\theta(\mathbf{d} \mid \mathbf{q})$
	$\mathbb{T}_\phi$	set of top-P documents ranked by f_ϕ
Sampling	$\mathbb{S}$	set of K documents sampled without replacement from $r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q})$
	$s(\mathbf{d})$	value of the priority weight for the document $\mathbf{d}$, $s(\mathbf{d}) = 0$ if $\mathbf{d} \notin \mathbb{S}$
	K	number of document samples with $K \leq P \leq N$
	C	number of Monte-Carlo samples (evaluation)
Bounds	$\log p_\theta(\mathbf{a}, \mathbf{q})$	Marginal task likelihood
	$\mathcal{L}_{\mathrm{VI}}(\mathbf{a}, \mathbf{q})$	Variational Lower bound (ELBO)
	$\mathcal{L}_\alpha(\mathbf{a}, \mathbf{q})$	Rényi Variational Bound (RVB)
	α	parameter of the RVB $\alpha \geq 0$
	$\hat{L}_\alpha^{\mathbb{S}}(\mathbf{a}, \mathbf{q})$	tractable RVB estimate given a set of $\mathbb{S}$ samples
	$L_\alpha^{K}(\mathbf{a}, \mathbf{q})$	standard RVB estimate given a set of K Monte-Carlo samples
	$D_{\mathrm{KL}}(r_\phi(\mathbf{d} \mid \mathbf{a}, \mathbf{q}) \Vert p_\theta(\mathbf{d} \mid \mathbf{a}, \mathbf{q}))$	KL divergence between the true and the approximate posteriors
Multiple-choice	$\mathbf{a}_i$	answer option i
	$\star$	index of the correct answer option
	$\mathbf{q}_i$	question-answer pair $[\mathbf{q}; \mathbf{a}_i]$
	M	number of answer options
	$\mathbf{A}$	vector of M answer choices
	$\mathbf{D}$	vector of M documents
	$\mathbf{Q}$	vector of M queries (each expressed as $[\mathbf{q}; \mathbf{a}_i]$)
	$g_\theta(\mathbf{d}, \mathbf{q})$	score of the reader (multiple-choice)
	$\mathbb{S}^{(M)}$	Cartesian product of the per-option samples $\mathbb{S}_1, \ldots, \mathbb{S}_M$
	$\boldsymbol{\mathcal{T}}_{\boldsymbol{\phi}}$	Product of the per-option top-P sets $\mathbb{T}_\phi(\mathbf{q}_1), \ldots, \mathbb{T}_\phi(\mathbf{q}_M)$
Spaces and Sets	Ω	space of strings
	$\mathbb{R}$	reals
	$(0, 1]$	real numbers in the interval [0, 1], 0 excluded
Operators	$:=$	defined as
	$[\cdot; \cdot]$	concatenation operator
	$\times$	Cartesian product
	$D_{\mathrm{KL}}(r_\phi(p \Vert q)$	Kullback–Leibler (KL) divergence between p and q
	$\mathbb{1}[\mathbf{x} \in \mathbb{X}]$	indicator function with value 1 if $\mathbf{x} \in \mathbb{X}$ otherwise 0
	$\mathrm{argtop}_{\mathbf{x} \in \mathbb{X}}(f(\mathbf{x}); K)$	top-K values from the set $\mathbb{X}$ as scored by f

G MEDWIKI

Table 11: Comparing the MedWiki with the original MedQA corpus on the USMLE dataset.

Method	Reader	Retriever	Corpus	Valid.	Test
Disjoint	BioBERT[1]	BM25	MedQA[2]	37.68	39.54
Disjoint	BioBERT[1]	BM25	MedWiki	38.82	40.46
Disjoint	BioLinkBERT	BM25	MedQA[2]	40.37	41.05
Disjoint	BioLinkBERT	BM25	MedWiki	**42.21**	**42.25**

[1]model weights from Lee et al. (2020), [2]original corpus from Jin et al. (2021)

The MedWiki corpus is a set of Wikipedia articles collected for research on medical question answering with low resources. Existing medical corpora, such as the MedQA corpus, are not adequately aligned with the ODQA task and are often measly and fragmented. At the same time, all of Wikipedia is cumbersome to use on consumer hardware. In order to reflect the true information need of medical experts, we assembled the MedWiki corpus by using real-world medical entrance exam questions. We queried the Wikipedia API using the answer options from all dataset splits of USMLE and MedMCQA and retained the top-10 articles for each answer option. This corpus includes 293.6k unique Wikipedia articles ($\approx 4.5\%$ of Wikipedia) that cover a broad range of medical topics.

MEDQA VS. MEDWIKI

Qualitative comparison When comparing knowledge corpora, there are two key questions: "how similar are they?" and "in what ways do they differ?". Using ElasticSearch, we compare the retrieved documents of MedWiki to the ones of MedQA. In Table 12, 13, 14 we present a few examples. The MedQA corpus is a selection of medical textbooks which often revolve around medical case studies, akin to the USMLE questions (see example in Table 12). In contrast, the MedWiki corpus references Wikipedia articles which are often edited to be concise, which is especially true for the abstract part of the articles, which contain the basic and usually most important information about a topic. Furthermore, each Wikipedia article comes with a title, which augments each passage with a higher-level context.

However, our approach of querying against the Wikipedia API results in many out-of-domain articles. For instance in Table 13, we display a MedWiki passage that originates from a non-medical article. Although the MedQA corpus is strictly oriented toward medical topics, it was built by extracting text from physical books using OCR software, which led to errors in the process and ultimately resulted in part of the corpus being unreadable.

Overall, both corpora provide adequate evidence to answer USMLE questions. Nevertheless, the MedWiki corpus is three times larger in vocabulary size and eight times more extensive in word count, making it more robust and diverse.

Quantitative comparison We investigated how the two corpora affect the final QA accuracy on the USMLE dataset. In contrast with the rest of the paper, we used a multi-document reader, as done in Jin et al. (2021). We used an ElasticSearch index to retrieve the set of top 3 documents $\{\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3\}$ for each pair $(\mathbf{q}, \mathbf{a}_i)$ as context for each answer option. The normalized log probabilities over the four options were obtained by processing the set of concatenated tokens $[\mathbf{d}_1; \mathbf{d}_2; \mathbf{d}_3; \mathbf{q}; \mathbf{a}_i]$ with BERT. We performed all experiments using a batch size of 16, set the learning rate to 1e-5, and run all experiments for 30 epochs. We report the predictive accuracy averaged for three initial random seeds.

Table 11 summarizes the performance on the two corpora. We see that our collected MedWiki corpus leads to better QA performance by 0.9%-1.2% absolute. This result indicates that the MedWiki corpus can safely be used as a replacement of the MedQA corpus. The MedWiki yields USMLE accuracy that is superior to using the MedQA corpus (Table 11), and yields good results on the MedMCQA (Table 4) despite consisting in only of a fraction of the English Wikipedia.

Question	a 5 year old girl is brought to the emergency department by her mother because of multiple episodes of nausea and vomiting that last about 2 hours. during this period she has had 6 8 episodes of bilious vomiting and abdominal pain. the vomiting was preceded by fatigue. the girl feels well between these episodes. she has missed several days of school and has been hospitalized 2 times during the past 6 months for dehydration due to similar episodes of vomiting and nausea. the patient has lived with her mother since her parents divorced 8 months ago. her immunizations are up to date. she is at the 60th percentile for height and 30th percentile for weight. she appears emaciated. her temperature is 36. 8 c 98. 8 f pulse is 99 min and blood pressure is 82 52 mm hg. examination shows dry mucous membranes. the lungs are clear to auscultation. abdominal examination shows a soft abdomen with mild diffuse tenderness with no guarding or rebound. the remainder of the physical examination shows no abnormalities. which of the following is the most likely diagnosis?
Options	**A: cyclic vomiting syndrome**, B: gastroenteritis, C: hypertrophic pyloric stenosis, D: gastroesophageal reflux disease
Document from MedQA	headache, and sweating patient presentation : be is a 45 - year - old woman who presents with concerns about sudden (paroxysmal), intense, brief episodes of headache, sweating (diaphoresis), and a racing heart (palpitations). focused history : be reports that the attacks started 3 weeks ago. they last from 2 to 10 minutes, during which time she feels quite anxious. during the attacks, it feels as though her heart is skipping beats (arrhythmia). at first, she thought the attacks were related to recent stress at work and maybe even menopause. the last time it happened, she was in a pharmacy and had her blood pressure taken. she was told it was 165 / 110 mm hg. be notes that she has lost weight (∼8 lbs) in this period even though her appetite has been good. pertinent findings : the physical examination was remarkable for be ' s thin, pale
Document from MedWiki	**panayiotopoulos syndrome**. pital, or calcarine sulci. follow - up meg demonstrated shifting localization or disappearance of meg spikes. illustrative cases in a typical presentation of panayiotopoulos syndrome, the child looks pale, vomits, and is fully conscious, able to speak, and understand but complains of " feeling sick. " two thirds of the seizures start in sleep ; the child may wake up with similar complaints while still conscious or else may be found vomiting, conscious, confused, or unresponsive. case 1. a girl had 2 seizures in sleep at 6 years of age. in the first fit she was found vomiting vigorously, eyes turned to one side, pale, and unresponsive. her condition remained unchanged for 3 hours before she developed generalized tonic - clonic convulsions. she gradually improved, and by the next morning was normal. the second seizure occurred 4 months later. she awoke and told her mother that she wanted to vomit,

Table 12: An example of the retrieved documents from the MedQA and MedWiki corpus respectively. Correct answers and document titles are highlighted when available.

Question	a 40 year old woman presents with difficulty falling asleep diminished appetite and tiredness for the past 6 weeks. she says that despite going to bed early at night she is unable to fall asleep. she denies feeling anxious or having disturbing thoughts while in bed. even when she manages to fall asleep she wakes up early in the morning and is unable to fall back asleep. she says she has grown increasingly irritable and feels increasingly hopeless and her concentration and interest at work have diminished. the patient denies thoughts of suicide or death. because of her diminished appetite she has lost 4 kg 8. 8 lb in the last few weeks and has started drinking a glass of wine every night instead of eating dinner. she has no significant past medical history and is not on any medications. which of the following is the best course of treatment in this patient?
Options	A: diazepam, B: paroxetine, C: zolpidem, **D: trazodone**
Document from MedQA	headache, and sweating patient presentation : be is a 45 - year - old woman who presents with concerns about sudden (paroxysmal), intense, brief episodes of headache, sweating (diaphoresis), and a racing heart (palpitations). focused history : be reports that the attacks started 3 weeks ago. they last from 2 to 10 minutes, during which time she feels quite anxious. during the attacks, it feels as though her heart is skipping beats (arrhythmia). at first, she thought the attacks were related to recent stress at work and maybe even menopause. the last time it happened, she was in a pharmacy and had her blood pressure taken. she was told it was 165 / 110 mm hg. be notes that she has lost weight (∼8 lbs) in this period even though her appetite has been good. pertinent findings : the physical examination was remarkable for be ' s thin, pale
Document from MedWiki	**hillary clinton's tenure as secretary of state**. hillary to the middle east to talk about how these countries can transition to new leaders — though, i've got to be honest, she's gotten a little passionate about the subject. these past few weeks it's been tough falling asleep with hillary out there on pennsylvania avenue shouting, throwing rocks at the window. in any case, obama's reference to clinton travelling a lot was true enough ; by now she had logged in her boeing 757, more than any other secretary of state for a comparable period of time, and had visited 79 countries while in the office. time magazine wrote that "clinton's endurance is legendary" and that she would still be going at the end of long work days even as her staff members were glazing out. the key was her ability to fall asleep on demand, at any time and place, for power naps. clinton also saw the potential political changes in the mideast as an opportunity for an even more fundamental change

Table 13: An example of the two different retrieved documents from the MedQA and MedWiki corpus. Correct answers and document titles are highlighted when available.

Question	a 37 year old female with a history of type ii diabetes mellitus presents to the emergency department complaining of blood in her urine left sided flank pain nausea and fever. she also states that she has pain with urination. vital signs include temperature is 102 deg f 39. 4 deg c blood pressure is 114 82 mmhg pulse is 96 min respirations are 18 and oxygen saturation of 97 on room air. on physical examination the patient appears uncomfortable and has tenderness on the left flank and left costovertebral angle. which of the following is the next best step in management?
Options	A: obtain an abdominal ct scan, **B: obtain a urine analysis and urine culture**, C: begin intravenous treatment with ceftazidime, D: no treatment is necessary
Document from MedQA	rim, & quinolones camille e. beauduy, pharmd, & lisa g. winston, md ⋆ a 59 - year - old woman presents to an urgent care clinic with a 4 - day history of frequent and painful urination. she has had fevers, chills, and flank pain for the past 2 days. her physician advised her to come immediately to the clinic for evaluation. in the clinic she is febrile (38. 5°c [101. 3°f]) but otherwise stable and states she is not experiencing any nausea or vomiting. her urine dipstick test is positive for leukocyte esterase. urinalysis and urine culture are ordered. her past medical history is significant for three urinary tract infections in the past year. each episode was uncom - plicated, treated with trimethoprim - sulfamethoxazole, and promptly resolved. she also has osteoporosis
Document from MedWiki	**hydronephrosis**. hydronephrosis describes dilation of the renal pelvis and calyces as a result of obstruction to urine flow. signs and symptoms the signs and symptoms of hydronephrosis depend upon whether the obstruction is acute or chronic, partial or complete, unilateral or bilateral. hydronephrosis that occurs acutely with sudden onset (as caused by a kidney stone) can cause intense pain in the flank area (between the hips and ribs). historically, this type of pain has been described as "dietl's crisis". conversely, hydronephrosis that develops gradually will generally cause either a dull discomfort or no pain. nausea and vomiting may also occur. an obstruction that occurs at the urethra or bladder outlet can cause pain and pressure resulting from distension of the bladder. blocking the flow of urine will commonly result in urinary tract infections which can lead to the development of stones, fever, and blood or pus in the urine

Table 14: An example of the two different retrieved documents from the MedQA and MedWiki corpus. Correct answers and document titles are highlighted when available.

BIBLIOGRAPHY

Abadi, Martín, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mane, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viegas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng (2016). "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems." In: arXiv: `1603.04467 [cs.DC]` (cit. on p. 11).

Baudiš (2015). "YodaQA: a modular question answering system pipeline." In: *ailao.eu* (cit. on p. 78).

Bender, Emily M, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell (2021). "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?" In: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. FAccT '21. Virtual Event, Canada: Association for Computing Machinery, pp. 610–623. ISBN: 9781450383097. DOI: `10.1145/3442188.3445922` (cit. on p. 77).

Borgeaud, Sebastian, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W Rae, Erich Elsen, and Laurent Sifre (2021). "Improving language models by retrieving from trillions of tokens." In: arXiv: `2112.04426 [cs.CL]` (cit. on pp. 79, 80).

Bornschein, Jörg and Yoshua Bengio (2015). "Reweighted Wake-Sleep." In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. URL: `http://arxiv.org/abs/1406.2751` (cit. on p. 48).

Bowman, Samuel R., Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio (2016). "Generating Sentences from a Continuous Space." In: *Pro-*

ceedings of The 20th SIGNLL Conference on Computational Natural Language Learning. Berlin, Germany: Association for Computational Linguistics, pp. 10–21. DOI: `10.18653/v1/K16-1002`. URL: `https://aclanthology.org/K16-1002` (cit. on pp. 1, 28, 35, 37).

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei (2020a). "Language Models are Few-Shot Learners." In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. URL: `https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html` (cit. on pp. 1, 30, 39, 70).

— (2020b). "Language Models are Few-Shot Learners." In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. URL: `https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html` (cit. on p. 67).

Burda, Yuri, Roger B. Grosse, and Ruslan Salakhutdinov (2016). "Importance Weighted Autoencoders." In: *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. URL: `http://arxiv.org/abs/1509.00519` (cit. on pp. 20, 49, 60).

Chen, Danqi, Adam Fisch, Jason Weston, and Antoine Bordes (2017a). "Reading Wikipedia to Answer Open-Domain Questions." In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, pp. 1870–1879. DOI: `10.18653/v1/P17-1171`. URL: `https://aclanthology.org/P17-1171` (cit. on pp. 78, 79).

Chen, Xi, Diederik P. Kingma, Tim Salimans, Yan Duan, Prafulla Dhariwal, John Schulman, Ilya Sutskever, and Pieter Abbeel (2017b). "Variational Lossy Autoencoder." In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net. URL: `https://openreview.net/forum?id=BysvGP5ee` (cit. on pp. 28, 36).

Chen, Xi, Nikhil Mishra, Mostafa Rohaninejad, and Pieter Abbeel (2018). "PixelSNAIL: An Improved Autoregressive Generative Model." In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 863–871. URL: `http://proceedings.mlr.press/v80/chen18h.html` (cit. on p. 36).

Child, Gray, Radford, and Sutskever (2019). "Generating long sequences with sparse transformers." In: *arXiv preprint arXiv:1904. 10509* (cit. on pp. 33, 36).

Child, Rewon (2021). "Very Deep VAEs Generalize Autoregressive Models and Can Outperform Them on Images." In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net. URL: `https://openreview.net/forum?id=RLRXCV6DbEJ` (cit. on pp. 34, 36).

Chowdhery, Aakanksha, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, et al. (2022). "PaLM: Scaling Language Modeling with Pathways." In: arXiv: `2204.02311 [cs.CL]` (cit. on p. 67).

Cremer, Chris, Xuechen Li, and David Duvenaud (2018). "Inference Suboptimality in Variational Autoencoders." In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 1086–1094. URL: `http://proceedings.mlr.press/v80/cremer18a.html` (cit. on p. 20).

Dempster, A P, N M Laird, and D B Rubin (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm." In: *Journal of the Royal Statistical Society. Series B, Statistical methodology* 39.1, pp. 1–38. ISSN: 1369-7412, 0035-9246 (cit. on p. 18).

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. DOI: `10.18653/v1/N19-1423`. URL: `https://aclanthology.org/N19-1423` (cit. on pp. 27, 30, 68, 79).

Dinh, Laurent, Jascha Sohl-Dickstein, and Samy Bengio (2017). "Density estimation using Real NVP." In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net. URL: `https://openreview.net/forum?id=HkpbnH9lx` (cit. on pp. 18, 27).

Dragusin, Radu, Paula Petcu, Christina Lioma, Birger Larsen, Henrik L Jørgensen, Ingemar J Cox, Lars Kai Hansen, Peter Ingwersen, and Ole Winther (2013). "FindZebra: A search engine for rare diseases." In: *International journal of medical informatics* 82.6, pp. 528–538. ISSN: 1386-5056. DOI: `10.1016/j.ijmedinf.2013.01.005` (cit. on p. 85).

Duffield, Nick, Carsten Lund, and Mikkel Thorup (2007). "Priority sampling for estimation of arbitrary subset sums." In: *Journal of the ACM* 54.6, 32–es. ISSN: 0004-5411. DOI: `10.1145/1314690.1314696` (cit. on p. 10).

Ferrucci, David, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A Kalyanpur, Adam Lally, J William Murdock, Eric Nyberg, John Prager, Nico Schlaefer, and Chris Welty (2010). "Building Watson: An Overview of the DeepQA Project." en. In: *AI Magazine* 31.3, pp. 59–79. ISSN: 2371-9621, 2371-9621. DOI: `10.1609/aimag.v31i3.2303` (cit. on p. 78).

Frostig, Roy, Matthew Johnson, and Chris Leary (2018). "Compiling machine learning programs via high-level tracing." en. In: *https://www.semanticscholar.org › paper › Compiling-ma...https://www.semanticscholar.org › paper › Compiling-ma...* (cit. on p. 11).

Ganguli, Deep, Danny Hernandez, Liane Lovitt, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova Dassarma, Dawn Drain, Nelson Elhage, Sheer El Showk, Stanislav Fort, Zac Hatfield-Dodds, Tom Henighan, Scott Johnston, Andy Jones, Nicholas Joseph, Jackson Kernian, Shauna Kravec, Ben Mann, Neel Nanda, Kamal Ndousse, Catherine Olsson, Daniela Amodei, Tom Brown, Jared Kaplan, Sam McCandlish, Christopher Olah, Dario Amodei, and Jack Clark (2022). "Predictability and Surprise in Large Generative Models." In: *2022 ACM Conference on Fairness, Accountability, and Transparency*. FAccT '22. Seoul, Republic of Korea: Association for Computing Machinery, pp. 1747–1764. ISBN: 9781450393522. DOI: `10.1145/3531146.3533229` (cit. on p. 68).

Gao, Leo, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy (2021). *The Pile: An 800GB Dataset of Diverse Text for Language Modeling*. DOI: `10.48550/ARXIV.2101.00027` (cit. on p. 77).

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (Nov. 2016). *Deep Learning*. en. MIT Press. ISBN: 9780262337373 (cit. on p. 27).

Grathwohl, Will, Dami Choi, Yuhuai Wu, Geoffrey Roeder, and David Duvenaud (2018). "Backpropagation through the Void: Optimizing control variates for black-box gradient estimation." In: *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net. URL: `https://openreview.net/forum?id=SyzKd1bCW` (cit. on p. 25).

Grathwohl, Will, Kuan-Chieh Wang, Jörn-Henrik Jacobsen, David Duvenaud, Mohammad Norouzi, and Kevin Swersky (2020). “Your classifier is secretly an energy based model and you should treat it like one.” In: *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020.* OpenReview.net. URL: `https://openreview.net/forum?id=Hkxzx0NtDB` (cit. on p. 27).

Grimmett, Geoffrey, Geoffrey R Grimmett, and David Stirzaker (2001). *Probability and Random Processes.* en. OUP Oxford. ISBN: 9780198572220 (cit. on p. 12).

Guu, Kelvin, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang (2020). “Retrieval Augmented Language Model Pre-Training.” In: *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event.* Vol. 119. Proceedings of Machine Learning Research. PMLR, pp. 3929–3938. URL: `http://proceedings.mlr.press/v119/guu20a.html` (cit. on pp. 41, 80, 81).

Higgins, Irina, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner (2017). “beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework.” In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings.* OpenReview.net. URL: `https://openreview.net/forum?id=Sy2fzU9gl` (cit. on p. 29).

Hinton, G E, P Dayan, B J Frey, and R M Neal (1995). “The “wake-sleep” algorithm for unsupervised neural networks.” en. In: *Science* 268.5214, pp. 1158–1161. ISSN: 0036-8075. DOI: `10.1126/science.7761831` (cit. on p. 48).

Hinton, Vinyals, and Dean (2015). “Distilling the knowledge in a neural network.” In: *arXiv preprint arXiv* (cit. on p. 85).

Ho, Jonathan, Xi Chen, Aravind Srinivas, Yan Duan, and Pieter Abbeel (2019). “Flow++: Improving Flow-Based Generative Models with Variational Dequantization and Architecture Design.” In: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA.* Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, pp. 2722–2730. URL: `http://proceedings.mlr.press/v97/ho19a.html` (cit. on p. 36).

Ho, Jonathan, Ajay Jain, and Pieter Abbeel (2020). “Denoising Diffusion Probabilistic Models.” In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.* Ed. by Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. URL: `https://proceedings.neurips.cc/paper/2020/hash/4c5bcfec8584af0d967f1ab10179ca4b-Abstract.html` (cit. on pp. 18, 27, 36).

Hochreiter, S and J Schmidhuber (1997). “Long short-term memory.” en. In: *Neural computation* 9.8, pp. 1735–1780. ISSN: 0899-7667. DOI: `10.1162/neco.1997.9.8.1735` (cit. on pp. 29, 36).

Hoogeboom, Emiel, Alexey A Gritsenko, Jasmijn Bastings, Ben Poole, Rianne van den Berg, and Tim Salimans (2021). “Autoregressive Diffusion Models.” In: arXiv: `2110.02037 [cs.LG]` (cit. on pp. 27, 30).

Izacard, Gautier, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave (2021). “Unsupervised Dense Information Retrieval with Contrastive Learning.” In: arXiv: `2112.09118 [cs.IR]` (cit. on p. 81).

Jang, Eric, Shixiang Gu, and Ben Poole (2017). “Categorical Reparameterization with Gumbel-Softmax.” In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net. URL: `https://openreview.net/forum?id=rkE3y85ee` (cit. on p. 46).

Jin, Di, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits (2021). “What disease does this patient have? A large-scale open domain question answering dataset from medical exams.” en. In: *APPS. Applied Sciences* 11.14, p. 6421. ISSN: 1454-5101, 2076-3417. DOI: `10.3390/app11146421` (cit. on p. 69).

Johnson, Jeff, Matthijs Douze, and Herve Jegou (2021). “Billion-scale similarity search with GPUs.” In: *IEEE transactions on big data* 7.3, pp. 535–547. ISSN: 2332-7790, 2372-2096. DOI: `10.1109/tbdata.2019.2921572` (cit. on p. 80).

Jordan, Michael I, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul (1999). “An Introduction to Variational Methods for Graphical Models.” In: *Machine learning* 37.2, pp. 183–233. ISSN: 0885-6125, 1573-0565. DOI: `10.1023/A:1007665907178` (cit. on pp. 17, 88).

Karpukhin, Vladimir, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih (2020). “Dense Passage Retrieval for Open-Domain Question Answering.” In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 6769–6781. DOI: `10.18653/v1/2020.emnlp-main.550`. URL: `https://aclanthology.org/2020.emnlp-main.550` (cit. on p. 80).

Kingma, Diederik P and Max Welling (2019). “An Introduction to Variational Autoencoders.” In: *Foundations and Trends® in Machine Learning* 12.4, pp. 307–392. ISSN: 1935-8237. DOI: `10.1561/2200000056` (cit. on p. 28).

Kingma, Diederik P. and Prafulla Dhariwal (2018). “Glow: Generative Flow with Invertible 1x1 Convolutions.” In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. Ed. by Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, pp. 10236–10245. URL: `https://proceedings.neurips.cc/paper/2018/hash/d139db6a236200b21cc7f752979132d0-Abstract.html` (cit. on pp. 27, 36).

Kingma, Diederik P., Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling (2014). "Semi-supervised Learning with Deep Generative Models." In: *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada.* Ed. by Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, pp. 3581–3589. URL: `https://proceedings.neurips.cc/paper/2014/hash/d523773c6b194f37b938d340d5d02232-Abstract.html` (cit. on p. 29).

Kingma, Diederik P. and Max Welling (2014a). "Auto-Encoding Variational Bayes." In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings.* Ed. by Yoshua Bengio and Yann LeCun. URL: `http://arxiv.org/abs/1312.6114` (cit. on pp. 13, 34).

— (2014b). "Auto-Encoding Variational Bayes." In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings.* Ed. by Yoshua Bengio and Yann LeCun. URL: `http://arxiv.org/abs/1312.6114` (cit. on p. 27).

Kingma, Durk P, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling (2016). "Improved Variational Inference with Inverse Autoregressive Flow." In: *Advances in Neural Information Processing Systems 29.* Ed. by D D Lee, M Sugiyama, U V Luxburg, I Guyon, and R Garnett. Curran Associates, Inc., pp. 4743–4751 (cit. on pp. 29, 34–36).

Kingma, Salimans, Poole, et al. (2021). "Variational diffusion models." In: *Advances in neural information processing systems.* ISSN: 1049-5258 (cit. on pp. 27, 36).

Kleijnen, Jack P C and Reuven Y Rubinstein (1996). "Optimization and sensitivity analysis of computer simulation models by the score function method." In: *European journal of operational research* 88.3, pp. 413–427. ISSN: 0377-2217. DOI: `10.1016/0377-2217(95)00107-7` (cit. on p. 13).

Kojima, Takeshi, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa (2022). "Large Language Models are Zero-Shot Reasoners." In: arXiv: `2205.11916 [cs.CL]` (cit. on pp. 1, 40, 67, 69, 70).

Kong (1992). "A note on importance sampling using standardized weights." In: *University of Chicago, Dept. of Statistics, Tech. Rep* (cit. on p. 9).

Kullback, S and R A Leibler (1951). "On Information and Sufficiency." In: *Annals of Mathematical Statistics* 22.1, pp. 79–86. ISSN: 0003-4851 (cit. on p. 19).

Lazaridou, Angeliki, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev (2022). “Internet-augmented language models through few-shot prompting for open-domain question answering.” In: arXiv: `2203.05115 [cs.CL]` (cit. on pp. 70, 78, 80).

Le, Tuan Anh, Adam R. Kosiorek, N. Siddharth, Yee Whye Teh, and Frank Wood (2019). “Revisiting Reweighted Wake-Sleep for Models with Stochastic Control Flow.” In: *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2019, Tel Aviv, Israel, July 22-25, 2019.* Ed. by Amir Globerson and Ricardo Silva. Vol. 115. Proceedings of Machine Learning Research. AUAI Press, pp. 1039–1049. URL: `http://proceedings.mlr.press/v115/le20a.html` (cit. on pp. 49, 54).

LeCun, Chopra, Hadsell, Ranzato, et al. (2006). “A tutorial on energy-based learning.” In: *Predicting structured* (cit. on p. 27).

Lee, Kenton, Ming-Wei Chang, and Kristina Toutanova (2019). “Latent Retrieval for Weakly Supervised Open Domain Question Answering.” In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.* Florence, Italy: Association for Computational Linguistics, pp. 6086–6096. DOI: `10.18653/v1/P19-1612`. URL: `https://aclanthology.org/P19-1612` (cit. on pp. 41, 79, 80).

Lewis, Patrick S. H., Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela (2020). “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.” In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.* Ed. by Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. URL: `https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html` (cit. on pp. 42, 80).

Li, Yingzhen and Richard E. Turner (2016). “Rényi Divergence Variational Inference.” In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain.* Ed. by Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, pp. 1073–1081. URL: `https://proceedings.neurips.cc/paper/2016/hash/7750ca3559e5b8e1f44210283368fc16-Abstract.html` (cit. on pp. 3, 22, 88).

Liévin, Valentin, Andrea Dittadi, Anders Christensen, and Ole Winther (2020). “Optimal Variance Control of the Score-Function Gradient Estimator for Importance-Weighted Bounds.” In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.* Ed. by Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. URL: `https://proceedings.`

neurips . cc / paper / 2020 / hash / c15203a83f778ce8934d0efaf2d5c6f3 - Abstract.html (cit. on pp. 45, 49, 59).

Liévin, Valentin, Andrea Dittadi, Lars Maaløe, and Ole Winther (2022a). “Towards Hierarchical Discrete Variational Autoencoders” (cit. on pp. 46, 47).

Liévin, Valentin, Andreas Geert Motzfeldt, Ida Riis Jensen Riis Jensen, and Ole Winther (2022b). “Variational Open-Domain Question Answering” (cit. on pp. 39, 59, 77, 80).

Liévin, Valentin, Christoffer Egeberg Hother, and Ole Winther (2022). “Can large language models reason about medical questions?” In: arXiv: 2207.08143 [cs.CL] (cit. on pp. 39, 67).

Liu, Pengfei, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig (2021). “Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing.” In: arXiv: 2107.13586 [cs.CL] (cit. on pp. 68, 89).

Maaløe, Lars, Marco Fraccaro, Valentin Liévin, and Ole Winther (2019). “BIVA: A Very Deep Hierarchy of Latent Variables for Generative Modeling.” In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada.* Ed. by Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett, pp. 6548–6558. URL: https://proceedings.neurips.cc/paper/2019/hash/9bdb8b1faffa4b3d41779bb495d79fb9-Abstract.html (cit. on pp. 33, 34, 36, 37).

Maddison, Chris J., Andriy Mnih, and Yee Whye Teh (2017). “The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables.” In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings.* OpenReview.net. URL: https://openreview.net/forum?id=S1jE5L5gl (cit. on p. 46).

Masrani, Vaden, Tuan Anh Le, and Frank Wood (2019). “The Thermodynamic Variational Objective.” In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada.* Ed. by Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett, pp. 11521–11530. URL: https://proceedings.neurips.cc/paper/2019/hash/618faa1728eb2ef6e3733645273ab145-Abstract.html (cit. on p. 56).

Maynez, Joshua, Shashi Narayan, Bernd Bohnet, and Ryan McDonald (2020). “On Faithfulness and Factuality in Abstractive Summarization.” In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* Online: Association for Computational Linguistics, pp. 1906–1919. DOI: 10.18653/v1/

2020.acl-main.173. URL: https://aclanthology.org/2020.acl-main.173 (cit. on pp. 69, 78).

Miao, Yishu and Phil Blunsom (2016). "Language as a Latent Variable: Discrete Generative Models for Sentence Compression." In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing.* Austin, Texas: Association for Computational Linguistics, pp. 319–328. DOI: 10.18653/v1/D16-1031. URL: https://aclanthology.org/D16-1031 (cit. on p. 39).

Mnih, Andriy and Karol Gregor (2014a). "Neural Variational Inference and Learning in Belief Networks." In: *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014.* Vol. 32. JMLR Workshop and Conference Proceedings. JMLR.org, pp. 1791–1799. URL: http://proceedings.mlr.press/v32/mnih14.html (cit. on p. 25).

— (2014b). "Neural Variational Inference and Learning in Belief Networks." In: *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014.* Vol. 32. JMLR Workshop and Conference Proceedings. JMLR.org, pp. 1791–1799. URL: http://proceedings.mlr.press/v32/mnih14.html (cit. on p. 55).

Mnih, Andriy and Danilo Jimenez Rezende (2016). "Variational Inference for Monte Carlo Objectives." In: *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016.* Ed. by Maria-Florina Balcan and Kilian Q. Weinberger. Vol. 48. JMLR Workshop and Conference Proceedings. JMLR.org, pp. 2188–2196. URL: http://proceedings.mlr.press/v48/mnihb16.html (cit. on pp. 14, 25, 48, 49, 55).

Mohamed, Shakir, Mihaela Rosca, Michael Figurnov, and Andriy Mnih (2019). "Monte Carlo Gradient Estimation in Machine Learning." In: arXiv: 1906.10652 [stat.ML] (cit. on p. 12).

Nowozin, Sebastian (2015). *Effective Sample Size in Importance Sampling.* http://www.nowozin.net/sebastian/blog/effective-sample-size-in-importance-sampling.html. Accessed: 2022-5-9 (cit. on p. 10).

Oord, Aäron van den, Nal Kalchbrenner, and Koray Kavukcuoglu (2016). "Pixel Recurrent Neural Networks." In: *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016.* Ed. by Maria-Florina Balcan and Kilian Q. Weinberger. Vol. 48. JMLR Workshop and Conference Proceedings. JMLR.org, pp. 1747–1756. URL: http://proceedings.mlr.press/v48/oord16.html (cit. on p. 36).

Oord, Aäron van den, Oriol Vinyals, and Koray Kavukcuoglu (2017). "Neural Discrete Representation Learning." In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA.* Ed. by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman

Garnett, pp. 6306–6315. URL: `https://proceedings.neurips.cc/paper/2017/hash/7a98af17e63a0ac09ce2e96d03992fbc-Abstract.html` (cit. on p. 46).

Ouyang, Long, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe (2022). "Training language models to follow instructions with human feedback." In: arXiv: `2203.02155 [cs.CL]` (cit. on p. 72).

Owen, Art B (2013). *Monte Carlo theory, methods and examples* (cit. on p. 10).

Pal, Ankit, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu (2022). "MedMCQA: A Large-scale Multi-Subject Multi-Choice Dataset for Medical domain Question Answering." In: *Proceedings of the Conference on Health, Inference, and Learning*. Ed. by Gerardo Flores, George H Chen, Tom Pollard, Joyce C Ho, and Tristan Naumann. Vol. 174. Proceedings of Machine Learning Research. PMLR, pp. 248–260 (cit. on pp. 69, 72).

Paranjape, Ashwin, Omar Khattab, Christopher Potts, Matei Zaharia, and Christopher D Manning (2021). "Hindsight: Posterior-guided training of retrievers for improved open-ended generation." In: arXiv: `2110.07752 [cs.CL]` (cit. on pp. 80, 81).

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala (2019). "PyTorch: An Imperative Style, High-Performance Deep Learning Library." In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*. Ed. by Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, pp. 8024–8035. URL: `https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html` (cit. on p. 11).

Radford, A, K Narasimhan, T Salimans, and I Sutskever (2018). "Improving language understanding by generative pre-training." In: *cs.ubc.ca* (cit. on p. 30).

Radford, Alec, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever (2019a). "Language Models are Unsupervised Multitask Learners." In: (cit. on p. 1).

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. (2019b). "Language models are unsupervised multitask learners." In: *OpenAI blog* 1.8, p. 9 (cit. on p. 30).

Rae, Jack W, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, et al. (2021). "Scaling Language Models: Methods, Analysis & Insights from Training Gopher." In: arXiv: `2112.11446 [cs.CL]` (cit. on p. 68).

Rainforth, Tom, Adam R. Kosiorek, Tuan Anh Le, Chris J. Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh (2018). "Tighter Variational Bounds are Not Necessarily Better." In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 4274–4282. URL: `http://proceedings.mlr.press/v80/rainforth18b.html` (cit. on pp. 14, 49–51, 54, 60, 61, 64, 88).

Rajpurkar, Pranav, Jian Zhang, Konstantin Lopyrev, and Percy Liang (2016). "SQuAD: 100,000+ Questions for Machine Comprehension of Text." In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics, pp. 2383–2392. DOI: `10.18653/v1/D16-1264`. URL: `https://aclanthology.org/D16-1264` (cit. on p. 78).

Ramesh, Aditya, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen (2022). "Hierarchical Text-Conditional Image Generation with CLIP Latents." In: arXiv: `2204.06125 [cs.CV]` (cit. on p. 1).

Rényi (1961). "On measures of entropy and information." In: *Proceedings of the fourth Berkeley symposium on* (cit. on p. 21).

Rezende, Danilo Jimenez, Shakir Mohamed, and Daan Wierstra (2014). "Stochastic Backpropagation and Approximate Inference in Deep Generative Models." In: *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*. Vol. 32. JMLR Workshop and Conference Proceedings. JMLR.org, pp. 1278–1286. URL: `http://proceedings.mlr.press/v32/rezende14.html` (cit. on pp. 13, 27).

Robertson, Stephen and Hugo Zaragoza (2009). "The Probabilistic Relevance Framework: BM25 and Beyond." In: *Foundations and Trends® in Information Retrieval* 3.4, pp. 333–389. ISSN: 1554-0669. DOI: `10.1561/1500000019` (cit. on p. 72).

Roeder, Geoffrey, Yuhuai Wu, and David Duvenaud (2017). "Sticking the Landing: Simple, Lower-Variance Gradient Estimators for Variational Inference." In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. Ed. by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, pp. 6925–6934. URL: `https://proceedings.neurips.cc/paper/2017/hash/`

e91068fff3d7fa1594dfdf3b4308433a-Abstract.html (cit. on pp. 14, 49, 53, 56, 63).

Sachan, Devendra Singh, Siva Reddy, William Hamilton, Chris Dyer, and Dani Yogatama (2021). "End-to-End Training of Multi-Document Reader and Retriever for Open-Domain Question Answering." en. In: *NeurIPS* (cit. on p. 81).

Saharia, Chitwan, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi (2022). *Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding*. DOI: 10.48550/ARXIV.2205.11487 (cit. on p. 1).

Salakhutdinov, Ruslan and Iain Murray (2008). "On the quantitative analysis of deep belief networks." In: *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*. Ed. by William W. Cohen, Andrew McCallum, and Sam T. Roweis. Vol. 307. ACM International Conference Proceeding Series. ACM, pp. 872–879. DOI: 10.1145/1390156.1390266. URL: https://doi.org/10.1145/1390156.1390266 (cit. on p. 55).

Salimans, Tim, Andrej Karpathy, Xi Chen, and Diederik P. Kingma (2017). "PixelCNN++: Improving the PixelCNN with Discretized Logistic Mixture Likelihood and Other Modifications." In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net. URL: https://openreview.net/forum?id=BJrFC6ceg (cit. on pp. 33, 36).

Semeniuta, Stanislau, Aliaksei Severyn, and Erhardt Barth (2017). "A Hybrid Convolutional Variational Autoencoder for Text Generation." In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 627–637. DOI: 10.18653/v1/D17-1066. URL: https://aclanthology.org/D17-1066 (cit. on pp. 35, 37).

Sennrich, Rico, Barry Haddow, and Alexandra Birch (2016). "Neural Machine Translation of Rare Words with Subword Units." In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, pp. 1715–1725. DOI: 10.18653/v1/P16-1162. URL: https://aclanthology.org/P16-1162 (cit. on p. 29).

Shannon, C E (1948). "A mathematical theory of communication." In: *The Bell System Technical Journal* 27.3, pp. 379–423. ISSN: 0005-8580. DOI: 10.1002/j.1538-7305.1948.tb01338.x (cit. on p. 19).

Shuster, Kurt, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston (2021). "Retrieval Augmentation Reduces Hallucination in Conversation." In: *Findings* *of the Association for Computational Linguistics: EMNLP 2021*. Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 3784–3803. DOI: `10.18653/v1/2021.findings-emnlp.320`. URL: `https://aclanthology.org/2021.findings-emnlp.320` (cit. on p. 78).

Singer, Uriel, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman (2022). "Make-A-Video: Text-to-Video Generation without Text-Video Data." In: arXiv: `2209.14792 [cs.CV]` (cit. on p. 1).

Sønderby, Casper Kaae, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther (2016). "Ladder Variational Autoencoders." In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*. Ed. by Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, pp. 3738–3746. URL: `https://proceedings.neurips.cc/paper/2016/hash/6ae07dcb33ec3b7c814df797cbda0f87-Abstract.html` (cit. on pp. 28, 34, 35).

Sønderby, Poole, and Mnih (2017). "Continuous Relaxation Training of Discrete Latent Variable Image Models." In: *Bayesian Deep Learning workshop, NIPS 2017* (cit. on p. 46).

Song, Xinying, Alex Salcianu, Yang Song, Dave Dopson, and Denny Zhou (2021). "Fast WordPiece Tokenization." In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 2089–2103. DOI: `10.18653/v1/2021.emnlp-main.160`. URL: `https://aclanthology.org/2021.emnlp-main.160` (cit. on p. 29).

Srivastava, Aarohi, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, et al. (2022). *Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models*. DOI: `10.48550/ARXIV.2206.04615` (cit. on p. 68).

Tucker, George, Dieterich Lawson, Shixiang Gu, and Chris J. Maddison (2019). "Doubly Reparameterized Gradient Estimators for Monte Carlo Objectives." In: *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net. URL: `https://openreview.net/forum?id=HkG3e205K7` (cit. on pp. 14, 49).

Tucker, George, Andriy Mnih, Chris J. Maddison, Dieterich Lawson, and Jascha Sohl-Dickstein (2017). "REBAR: Low-variance, unbiased gradient estimates for discrete latent variable models." In: *Advances in Neural Information Processing Systems 30:*

Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA. Ed. by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, pp. 2627–2636. URL: `https://proceedings.neurips.cc/paper/2017/hash/ebd6d2f5d60ff9afaeda1a81fc53e2d0-Abstract.html` (cit. on p. 25).

Vahdat, Arash and Jan Kautz (2020). "NVAE: A Deep Hierarchical Variational Autoencoder." In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin. URL: `https://proceedings.neurips.cc/paper/2020/hash/e3b21256183cf7c2c7a66be163579d37-Abstract.html` (cit. on pp. 34, 36).

Vahdat, Arash, William G. Macready, Zhengbing Bian, Amir Khoshaman, and Evgeny Andriyash (2018). "DVAE++: Discrete Variational Autoencoders with Overlapping Transformations." In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 5042–5051. URL: `http://proceedings.mlr.press/v80/vahdat18a.html` (cit. on p. 36).

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin (2017). "Attention is All you Need." In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. Ed. by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, pp. 5998–6008. URL: `https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html` (cit. on p. 29).

Vieira, Tim (2017). *Estimating means in a finite universe*. `https://timvieira.github.io/blog/post/2017/07/03/estimating-means-in-a-finite-universe/`. Accessed: 2022-NA-NA (cit. on p. 11).

Wang, Xuezhi, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou (2022). *Self-Consistency Improves Chain of Thought Reasoning in Language Models*. DOI: `10.48550/ARXIV.2203.11171` (cit. on pp. 70, 71).

Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou (2022). "Chain of Thought Prompting Elicits Reasoning in Large Language Models." In: arXiv: `2201.11903 [cs.CL]` (cit. on pp. 68, 69).

Williams, Ronald J (1992). “Simple statistical gradient-following algorithms for connectionist reinforcement learning.” In: *Machine learning* 8.3, pp. 229–256. ISSN: 0885-6125, 1573-0565. DOI: `10.1007/BF00992696` (cit. on p. 13).

Yang, Zhilin, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le (2019). “XLNet: Generalized Autoregressive Pretraining for Language Understanding.” In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*. Ed. by Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d’Alché-Buc, Emily B. Fox, and Roman Garnett, pp. 5754–5764. URL: `https://proceedings.neurips.cc/paper/2019/hash/dc6a7e655d7e5840e66733e9ee67cc69-Abstract.html` (cit. on pp. 30, 33).

Yasunaga, Michihiro, Jure Leskovec, and Percy Liang (2022). “LinkBERT: Pretraining Language Models with Document Links.” In: arXiv: `2203.15827 [cs.CL]` (cit. on pp. 72, 84).

Zhu, Yukun, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler (2015). “Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books.” In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, pp. 19–27. DOI: `10.1109/ICCV.2015.11`. URL: `https://doi.org/10.1109/ICCV.2015.11` (cit. on p. 35).