Posterior Collapse in Deep Latent Variable Modeling of Text

Shaine Leibowitz

Abstract

The Variational Autocoder (VAE) can be a powerful tool for language modeling and representation learning. The generative model extends the RNNLM by incorporating latent representations of sentences. This provides us with the following advantages: a greater control over features such as style, topic, and high-level syntax as well as decoding diverse and plausible sentences by sampling from the prior. However, effectively estimating the *evidence lower bound* (ELBO) to the intractable log marginal likelihood can be challenging. When applying autoregressive decoders, a trivial local optimum of the surrogate objective is reached. This phenomenon is known as *posterior collapse*. This paper implements the combination of pretraining of the inference network on the Autoencoder objective and training with a constraint on the minimum amount of information encoded into the latent variables. When compared to linear KL annealing or pretraining on its own, these heuristics show a significant improvement in language modeling and latent representation learning metrics. Additionally, this paper strives to add to the dialogue regarding whether the ELBO is a proper objective for deep latent variable modeling of text. [1]

1. Introduction

Probabilistic models having continuous latent variables $\mathbf{z}$ seek to find low-dimensional representation of the observations $\mathbf{x}$. One powerful framework for representation learning is the Variational Autoencoder (VAE) (Kingma & Welling, 2013).

Though powerful, VAEs encounter the *posterior collapse* problem in the context of modeling text (Bowman et al., 2015b). In modeling text sequences where $\mathbf{x}^{(i)} = x_1, ..., x_T$, autoregressive decoders such as LSTMs tend to be implemented due to the long-term memory requirements of language. The usage of these strong decoders often cause training to yield a detrimental local optimum to the *evidence lower bound* (ELBO) as the decoder learns to disregard the latent variables during reconstruction. Therefore, the encoder does not encode information from the observations thereby collapsing the posterior distribution to the prior. When this happens, the model essentially reduces to a language model. Motivations for latent representation modeling include providing more diverse samples and more control over features such as style and author's voice.

The prominent methods for tackling posterior collapse include re-weighting the loss function (Bowman et al., 2015b; Kingma et al., 2017; Pelsmaeker & Aziz, 2019; Chen et al., 2016), pretraining on a different objective (Li et al., 2019), and weakening the decoder (Bowman et al., 2015b; Semeniuta et al., 2017).

This paper adheres to the techniques of (Li et al., 2019) by pretraining the inference network using an autoencoder objective, linear annealing at the beginning of training (Bowman et al., 2015b), and replacing the regularization term (KL) of the objective function (ELBO) (Kingma et al., 2017). (Li et al., 2019) demonstrated that these technique perform better together than in isolation. I also experiment with a smoother version of the objective function so that the KL term is weighted instead of replaced (Chen et al., 2016). This provides the benefits of smooth gradients and a greater alignment with the ELBO.

A major recommendation from (Chen et al., 2016) is to reconsider using ELBO as the surrogate objective in future research. They draw this conclusion since their method achieved better language modeling results measured by perplexity but a lower ELBO value than other leading methods. It follows logically that their method which replaces a part of the ELBO would result in an inferior ELBO. This paper explores if a smoother objective function obtains strong language modeling and ELBO results to further the discussion of whether the ELBO is an appropriate surrogate objective.

2. Background

2.1. Variational Autoencoder

It is believed that in many cases, observations can be explained by interesting features having much lower dimen-

[1] Code is available at `https://github.com/shainedl/Papers-Colab/blob/master/Posterior_Collapse.ipynb`

sionality than that of the original data space. What cannot be explained by the latent variables can be attributed to noise.

Autoencoders encode information from the observations to find this low-dimensional representation, and the decoder then reconstructs the observations from the latent variables. Adapted from the Autoencoder architecture, Variational Autoencoders learn an approximate posterior distribution with the use of Variational Inference (VI).

By replacing the deterministic function of Autoencoders with an approximate posterior distribution, VAEs map representations to smooth regions in latent space. Since the VAE is regularized to push the posterior distribution towards the prior, decoding any point in the latent space with a reasonable probability under the prior will result in a plausible sentence. The prior distribution is commonly a standard Gaussian.

2.2. Data

The dataset is downsampled from The Stanford Natural Language Inference (SNLI) Corpus (Bowman et al., 2015a). The SNLI corpus is a collection of 570k human-written English sentence pairs. The corpus was intended to serve both as a benchmark for evaluating representational systems for text, especially including those induced by representation learning methods.

3. Related Work

(Bowman et al., 2015b) introduced the variational autoencoder architecture for text and its obstacle of posterior collapse. Their proposed solutions centered around reweighting the KL term and weakening the decoder. In the beginning epochs, the weight increases linearly from 0 to 1. This can be thought of as annealing from a vanilla autoencoder to a variational autoencoder. (Bowman et al., 2015b) also experiment with word dropout and historyless decoding. By limiting the number of words the decoder can condition on, the decoder has to rely more on the latent variables.

Since the original identification of the problem, there has been a plethora of solutions proposed. (Pelsmaeker & Aziz, 2019) sought to provide a comprehensive comparison of the leading methods such as the Free Bits method (Kingma et al., 2017) which discourages the model from decreasing the KL term given a threshold. (Pelsmaeker & Aziz, 2019) demonstrates that Free Bits outperforms the annealing and word dropout techniques of (Bowman et al., 2015b). (Li et al., 2019) use these findings as justification to center their methodology around Free Bits. (Li et al., 2019) combine the Free Bits (FB) objective, pretraining (PT) the encoder using the Autoencoder objective, and linear annealing (KA) for the first 10 epochs. (Bowman et al., 2015b), (Pelsmaeker & Aziz, 2019), and (Li et al., 2019) all use the Penn Treebank as the source of their data. (Li et al., 2019) additionally use data from SNLI and Yahoo.

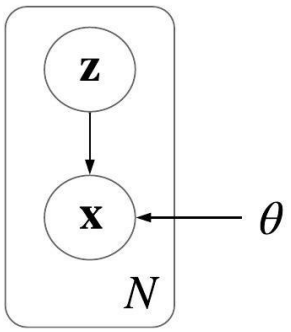

Figure 1. The directed graphical model under consideration. The lines denote the generative model $p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})p(\mathbf{z})$. $\boldsymbol{\theta}$ are the generative model parameters.

Though (Pelsmaeker & Aziz, 2019) show that Free Bits performs relatively well, they support methods that reweight rather than replace the KL term such as Soft Free Bits (SFB) (Chen et al., 2016) and their own approach Minimum Desired Rate. The methodology in this paper closely follows that of (Li et al., 2019). I experiment with their procedure of pretraining, KL linear annealing, and Free Bits objective for training. The main contribution of this paper is testing the objective Soft Free Bits after pretraining and KL linear annealing.

4. Model

Let us consider the following deep generative model with parameters $\boldsymbol{\theta}$. With observations $\mathbf{x} = \mathbf{x}^{(1)}, ..., \mathbf{x}^{(N)}$ and continuous m-dimensional latent variables $\mathbf{z} = z_1, ..., z_m$, we can specify sampling from the joint distribution. The graphical model can be seen in Figure 1.

$$\tilde{\mathbf{z}} \sim p(\mathbf{z}) \qquad \tilde{\mathbf{x}} \sim p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})$$

In order to determine the log likelihood of the observations, the latent variable is marginalized out.

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}) = \log \int_{\mathbf{z}} p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

However, this integral, and consequently posterior inference, is usually intractable to compute.

5. Inference (or Training)

Since the marginal log likelihood is intractable to compute, we apply the Amortized Variational Inference method to

approximate the posterior distribution with variational parameters ϕ.

$$\tilde{\mathbf{z}} \sim q_{\phi}(\mathbf{z}|\mathbf{x}) \qquad \tilde{\mathbf{x}} \sim p_{\boldsymbol{\theta}}(\mathbf{x}|\tilde{\mathbf{z}})$$

We would like the approximate distribution to be as close as possible to the posterior distribution. Therefore, the objective is as follows

$$\arg\min_{\phi} \mathrm{KL}(q_{\phi}(\mathbf{z}|\mathbf{x})||p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}))$$

Minimizing the above KL divergence can be shown to be equivalent to maximizing the following objective. Additionally, this provides a lower bound on the marginal log likelihood.

$$\begin{aligned}\log p_{\boldsymbol{\theta}}(\mathbf{x}) &\geq \mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})}[\log(p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}))] - \mathrm{KL}(q_{\phi}(\mathbf{z}|\mathbf{x})||p_{\boldsymbol{\theta}}(\mathbf{z})) \\ &= \mathrm{ELBO}(\boldsymbol{\theta}, \phi, \mathbf{x})\end{aligned}$$

The first term of the ELBO serves as the negative reconstruction error while the second term acts as a regularizer to keep the posterior distribution close to the prior. The variational autoencoder learns the parameters of the inference network jointly with the global model parameters. When $\mathrm{KL}(q_{\phi}(\mathbf{z}|\mathbf{x})||p_{\boldsymbol{\theta}}(\mathbf{z})) \approx 0$, the posterior has collapsed.

Algorithm 1 Pseudocode for Pretraining and Training with Soft Free Bits

Input:
$\mathbf{x}$, inference network ϕ, generative model $\boldsymbol{\theta}$, learning rate α, target rate λ, anneal weight γ, threshold β, multiplier χ, loss function $f(\boldsymbol{\theta}, \phi, \mathbf{x})$
Procedure:
$[\boldsymbol{\mu}, \boldsymbol{\sigma}] \leftarrow \mathrm{enc}(\mathbf{x}; \phi)$
$\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$
$\mathbf{z} \leftarrow \boldsymbol{\sigma} \odot \boldsymbol{\epsilon} + \boldsymbol{\mu}$
$\mathbf{o} \leftarrow \mathrm{dec}(\mathbf{x}, \mathbf{z}; \boldsymbol{\theta})$
RE $\leftarrow$ CrossEntropyLoss($\mathbf{o}$, $\mathbf{x}$)
KL $\leftarrow 0.5 * [\boldsymbol{\mu}^2 + \boldsymbol{\sigma}^2 - 1 - \log(\boldsymbol{\sigma}^2)]$
$f(\boldsymbol{\theta}, \phi, \mathbf{x}) \leftarrow$ RE + γKL
while pretraining has not converged **do**
  $\gamma \leftarrow 0$
end while
for epoch in 1 to 10 **do**
  $\gamma \leftarrow \gamma + 0.1$
end for
while SFB training has not converged **do**
  if KL $> (1.0 + \beta) * \lambda$ **then**
    $\gamma \leftarrow \min(\gamma * (1.0 + \chi), 1.0)$
  else if KL $< (1.0 - \beta) * \lambda$ **then**
    $\gamma \leftarrow \gamma * (1.0 - \chi)$
  end if
end while
Update θ, ϕ based on $\frac{\partial \mathcal{L}}{\partial \theta}$, $\frac{\partial \mathcal{L}}{\partial \phi}$

6. Methods

6.1. Surrogate Objectives

In the Free Bits method, the KL term is replaced with the following term so that each dimension of the KL term is at least a target rate λ:

$$\sum_{i} \max[\lambda, \mathrm{KL}(q_{\phi}(\mathbf{z}|\mathbf{x})||p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}))]$$

In KL annealing, a parameter re-weights the KL term in the ELBO objective:

$$\mathbb{E}_{q_{\phi}(\mathbf{z}|\mathbf{x})}[\log(p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}))] - \gamma \mathrm{KL}(q_{\phi}(\mathbf{z}|\mathbf{x})||p_{\boldsymbol{\theta}}(\mathbf{z}))$$

With linear KL annealing, the anneal rate γ linearly increases from 0 to 1 over a certain number of epochs. The method where γ increases linearly in the first 10 epochs without any prior Autoencoder objective pretraining (KLA) is used as the baseline in this paper. Other methods that are evaluated in order to compare their resulting metrics include the vanilla Autoencoder objective by itself as well as the combination of pretraining and KL annealing.

The γ in Soft Free Bits is tuned according to the target rate λ of bits and is updated online. If the KL for the epoch is at least a threshold β above λ, then γ is increased by χ. On the other hand, if the KL for the epoch is at least a threshold β below λ, then γ is decreased by χ to allow encoding of more information.

6.2. Experimental Setup

The dataset was randomly downsampled from SNLI consisting of 60K/6K/6K sentences for training/validation/test, and the sentence size was limited to 12 tokens for a lower computational cost. The details of the experimental setup follow (Li et al., 2019). Both the encoder and decoder used a one-layer LSTM with the same hidden size. The LSTM parameters and embedding are initialized with uniform distribution: $\mathcal{U}(-0.01, 0.01)$ and $\mathcal{U}(-0.1, 0.1)$ respectively.

Word Embedding Size	Hidden Size	Latent Dim m
128	512	32

A dropout was applied in the decoder to word embeddings as well as the last output from the LSTM. The SGD optimizer is used without momentum during training, If validation loss does not improve, the learning rate decays, and training stops early after 5 learning rate decays. Pretraining ran for 3 hours, Training with KL Annealing without prior pretraining ran for 1.25 hours and with pretraining ran for 2.75 hours, Training with Free Bits ran for 1.5 hours, and Training with Soft Free Bits ran for 2.25 hours using the GPUs from Google Colab with a batch size of 4.

Method	NLL	PPL	RE	KL	-ELBO	ELBO PPL
AE			27.07			
KLA	37.12	49.12	36.04	0.02	36.06	43.81
PT + KLA	37.38	50.44	36.27	0.01	36.27	44.81
PT + KLA + FB	36.50	45.99	31.97	4.84	36.81	47.40
PT + KLA + SFB	36.75	47.25	32.44	4.29	36.73	46.99

Table 1. Language modeling results on SNLI test set.

Dropout p	Learning Rate α	Decay Rate	Patience
0.5	0.5 (Initialized)	0.5	2

For Free Bits and Soft Free Bits training, λ is not tuned as (Li et al., 2019) note that their method is not sensitive to its changes. (Pelsmaeker & Aziz, 2019) argue against the Soft Free Bits method due to the extra need for hyperparameter tuning. By setting the hyperparameters to those of (Chen et al., 2016), SFB maintains the simplicity of FB.

Target Rate λ	Threshold β	Multiplier χ	Constraint
4	0.05	0.1	$0 \leq \gamma \leq 1$

6.3. Metrics

The quantitative evaluation focuses on the metrics perplexity per token (PPL) and the negative ELBO. PPL is calculated from the negative log likelihood (NLL) which is approximated from estimating the log marginal likelihood with 500 importance weighted samples (Li et al., 2019). Importance sampling leads to a tighter bound and lower variance. With the -ELBO calculation, we look at the reconstruction error (RE) and Kullback-Leibler divergence (KL) terms separately. The goals are to have a low PPL, -ELBO, and RE. Ideally for KL, we are looking for a small but non-zero result which is captured in the target rate λ.

The qualitative analysis concentrates on linear interpolation between latent variables. As exemplified by (Bowman et al., 2015b), the latent space is smooth if the decoded sentences between samples from $p(\mathbf{z})$ are plausible.

7. Results

The test set results can be seen in Table 1. Training on only the vanilla Autoencoder unsurprisingly brings about the smallest reconstruction error since the Autoencoder's only objective is to minimize reconstruction error. The KLA and PT+KLA baselines both clearly suffer from posterior collapse as both of their KL terms are near 0. They have higher reconstruction errors but lower -ELBOs due to the near-zero KL terms. They have the worst importance sampling derived PPLs.

As hypothesized, training with Soft Free Bits amounted to a lower -ELBO than with Free Bits. Though PT+KLA+SFB had a higher reconstruction error than PT+KLA+FB, training with Soft Free Bits led to a lower KL term that was closer to the target rate λ. Overall, PT+KLA+FB has the lowest importance sampling estimate of PPL.

We can see the greedily decoded sentences in Table 2 where the first sentence and last sentences are sampled from the prior distribution. In between, the sentences are greedily decoded from linear interpolation between the two points in latent space. In the training with the Autoencoder objective, all of the sentences are the least syntactically sensible or grammatical. In the KL Annealing output, the sampled sentences are plausible except for an additional comma. However, the posterior collapse creates an incapacity to interpolate and splits the decoding between the two samples. Though not shown, the posterior collapse in PT+KLA caused a similar phenomenon (for this and more examples of interpolations from each method, please see the github).

The Free Bits and Soft Free Bits methods both have grammatical and mostly plausible sentences through the entire interpolation exhibiting a smooth latent space. There are a couple sentences in each interpolation that do not completely make sense. In the second-to-last sentence in PT+KLA+FB, a woman is 'holding a baby in her mouth' which I hope would not happen in actuality! For a few sentences PT+KLA+SFB, a man is holding a baby and a weed-eater at the same time which is possible but not a likely scenario. Other than that, their interpolated sentences perform well in terms of plausibility. Another important observation is that both of these methods provide diversity in the sentences sampled from the prior. On the other hand, the sentences sampled from the prior when the posterior collapsed were the same with the exception of a one word difference: the word 'shirt' became 'hat' in the second sample.

8. Discussion

My previous hypothesis was that considering the SFB objective is more closely aligned with the ELBO objective that it would perform better than the FB method. This was true in terms of the ELBO performance and nearness with the target rate λ. However, the method incorporating SFB as well as the baselines all had an inferior importance sampling PPL but a superior ELBO. This is the reasoning be-

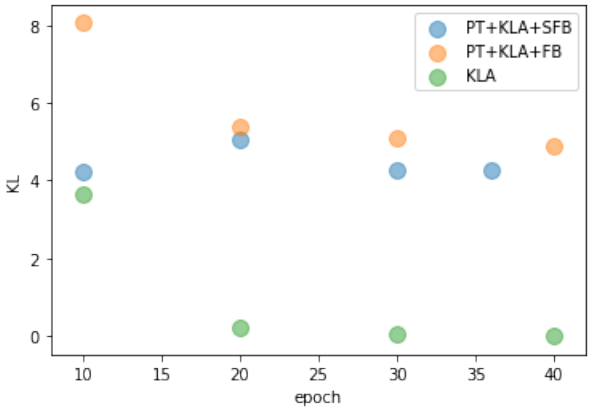

Figure 2. The Validation KL term per epoch. We can compare the different methods to the Baseline over time.

hind the claim in (Li et al., 2019) that the ELBO is not an appropriate metric in this context.

We were also able to see the effects of posterior collapse. In the interpolation between sentences, posterior collapse resulted in sentences lacking diversity throughout the latent space.

Let's take a look at how the KL term changed over time in Figure 2. Towards the end of the KL annealing, the KL term in KLA was appropriate and around that of the method with SFB. The KL term quickly drops due to posterior collapse and hovers around 0 for the remaining epochs. The method with FB starts with out the highest KL term, then takes a dip and is persistently mildly above the SFB method. The SFB method is the most stable and is consistently around the target rate λ. This points to the smoother nature of Soft Free Bits over Free Bits.

As mentioned before, the hyperparameters for Soft Free Bits were not tuned specifically for this model. The purpose of this was to combat a major argument against the Soft Free Bits method that it is overly complex due to tuning. So, it is helpful to reiterate that favorable results came from an untuned model. This is of course not to discourage tuning hyperparameters in practice.

9. Conclusion

In this paper, we observe how an amalgam of techniques provides a strong defense against posterior collapse in VAEs. As ascertained, the combination of pretraining and linear KL annealing improves variants of the Free Bits method. Future research can help to uncover the influence of pretraining on other objective functions and the role of ELBO in training and evaluating deep latent variable modeling of text.

Autoencoder
unk-initCap teacher performs basketball
unk-initCap teacher performs basketball
unk-initCap teacher performs with
unk-initCap teacher giving a
Boy 's has a ball
Boy 's has a kids
Boy 's has a bird
Man 's time a store
Person 's making a store .
Person with ready a field .
Person with ready a field .
KL Annealing
A man in a blue shirt , and a black shirt smiles .
A man in a blue shirt , and a black shirt smiles .
A man in a blue shirt , and a black shirt smiles .
A man in a blue shirt , and a black shirt smiles .
A man in a blue shirt , and a black shirt smiles .
A man in a blue shirt , and a black shirt smiles .
A man in a blue shirt , and a black shirt smiles .
A man in a blue shirt , and a black shirt smiles .
A man in a blue shirt , and a black shirt smiles .
A man in a blue shirt , and a black hat smiles .
A man in a blue shirt , and a black hat smiles .
Pretraining, KL Annealing, Free Bits
A man in a blue shirt is running through a field of grass .
A man is walking in a field , holding a stick in thought .
A man is walking in a field , holding a stick in thought .
A man is standing in front of a large body of water .
A man is standing in front of a large building , possibly .
A man is standing in front of a large building , possibly .
A man is standing in front of a large building , possibly .
A man is standing in front of a large building , holding a bottle .
A woman is walking down a slide , holding a camera in her mouth .
A woman is walking down a street , holding a baby in her mouth .
A woman is walking down the street , holding a baby .
Pretraining, KL Annealing, Soft Free Bits
A man is looking at the camera , while others watch the camera .
A man is looking at the camera , while others watch the camera .
A man is holding a baby , while he is holding a weed-eater .
A man is holding a baby , while he is holding a weed-eater .
A man is holding a baby , while he is holding a weed-eater .
A man is holding a baby , while he is holding a weed-eater .
Two men are standing in front of a large crowd of people .
Two men are playing with a toy in the snow , and they wait .
Two men are playing with a toy in the snow , and they wait .
Two men are playing with a toy in the snow , and they are outside .
Two men are playing with a toy in the snow , and they are outside .

Table 2. Interpolation between prior samples

References

Bowman, Samuel R., Angeli, Gabor, Potts, Christopher, and Manning, Christopher D. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2015a.

Bowman, Samuel R., Vilnis, Luke, Vinyals, Oriol, Dai, Andrew M., Józefowicz, Rafal, and Bengio, Samy. Generating sentences from a continuous space. In *CoNLL*, 2015b.

Chen, Xi, Kingma, Diederik P., Salimans, Tim, Duan, Yan, Dhariwal, Prafulla, Schulman, John, Sutskever, Ilya, and Abbeel, Pieter. Variational lossy autoencoder. *ArXiv*, abs/1611.02731, 2016.

Kingma, Diederik P. and Welling, Max. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013.

Kingma, Diederik P., Salimans, Tim, and Welling, Max. Improved variational inference with inverse autoregressive flow. *ArXiv*, abs/1606.04934, 2017.

Li, Bohan, He, Junxian, Neubig, Graham, Berg-Kirkpatrick, Taylor, and Yang, Yiming. A surprisingly effective fix for deep latent variable modeling of text. *ArXiv*, abs/1909.00868, 2019.

Pelsmaeker, Tom and Aziz, Wilker. Effective estimation of deep generative language models. *ArXiv*, abs/1904.08194, 2019.

Semeniuta, Stanislau, Severyn, Aliaksei, and Barth, Erhardt. A hybrid convolutional variational autoencoder for text generation. In *EMNLP*, 2017.