SentenceMIM: A Latent Variable Language Model

Micha Livne 1 2 Kevin Swersky 3 David J. Fleet 1 2

Abstract

We introduce sentenceMIM, a probabilistic auto-encoder for language modelling, trained with Mutual Information Machine (MIM) learning. Previous attempts to learn variational auto-encoders for language data have had mixed success, with empirical performance well below state-of-the-art auto-regressive models, a key barrier being the occurrence of posterior collapse with VAEs. The recently proposed MIM framework encourages high mutual information between observations and latent variables, and is more robust against posterior collapse. This paper formulates a MIM model for text data, along with a corresponding learning algorithm. We demonstrate excellent perplexity (PPL) results on several datasets, and show that the framework learns a rich latent space, allowing for interpolation between sentences of different lengths with a fixed-dimensional latent representation. We also demonstrate the versatility of sentenceMIM by utilizing a trained model for question-answering, a transfer learning task, without fine-tuning. To the best of our knowledge, this is the first latent variable model (LVM) for text modelling that achieves competitive performance with non-LVM models.

1. Introduction

Generative modelling of text has become one of the predominant approaches to natural language processing (NLP), particularly in the machine learning community. It is favoured because it supports probabilistic reasoning and it provides a principled framework for unsupervised learning in the form of maximum likelihood. Unlike computer vision, where various generative approaches have proliferated (Dinh et al., 2017; Goodfellow et al., 2014; Kingma & Welling, 2013; Oord et al., 2016; Rezende et al., 2014), current methods for text mainly rely on auto-regressive models.

Generative latent variable models (LVMs), such as the variational auto-encoder (VAE) (Kingma & Welling, 2013; Rezende et al., 2014), are versatile and have been successfully applied to a myriad of domains. Such models consist of an encoder, which maps observations to distributions over latent codes, and a decoder that maps latent codes to distributions over observations. LVMs are widely used and studied because they can learn a latent representation that carries many useful properties. Observations are encoded as fixed-length vectors that capture salient information, allowing for semantic comparison, interpolation, and search. They are often useful in support of downstream tasks, such as transfer or k-shot learning. They are also often interpretable, capturing distinct factors of variation in different latent dimensions. These properties have made LVMs especially compelling in the vision community.

Despite their desirable qualities, generative LVMs have not enjoyed the same level of success in text modelling. There have been several recent proposals to adapt VAEs to text (Bowman et al., 2015; Guu et al., 2017; Kruengkrai, 2019; Li et al., 2019b; Yang et al., 2017), but despite encouraging progress, they have not reached the same level of performance on natural language benchmarks as auto-regressive models (e.g., (Merity et al., 2017; Rae et al., 2018; Wang et al., 2019)). This is often attributed to the phenomenon of posterior collapse (Le Fang, 2019; Li et al., 2019a), in which the decoder captures all of the modelling power and the encoder ends up conveying little to no information. For text, where the decoder is naturally auto-regressive, this has proven challenging to mitigate.

This paper introduces sentenceMIM (sMIM), a new LVM for text. It is based on the architecture of Bowman et al. (2015) and the mutual information machine (MIM) framework (Livne et al., 2019). MIM is a recently introduced LVM framework that shares the same underlying architecture as VAEs, but uses a different learning objective that is more robust against posterior collapse. MIM learns a highly informative and compressed latent representation, and often strictly benefits from more powerful architectures. To evaluate sMIM we propose a novel bound on the model log-likelihood, called MIM-ELBO, or MELBO. As an alternative to the evidence lower bound (ELBO) used to evaluate
VAEs, MELBO is useful for models with implicit priors, for which the ELBO is intractable.

We show on four challenging datasets that sMIM outperforms VAE models for text, and is competitive with state-of-the-art auto-regressive approaches, including transformer-based models (Radford et al., 2019; Vaswani et al., 2017), as measured by negative log-likelihood and perplexity. We further demonstrate the quality of the sMIM representation by generating diverse samples around a given sentence and interpolating between sentences. Finally, we show the versatility of the learned representation by applying a pre-trained sMIM model to a question answering task with state-of-art performance as compared to single task, supervised models.

2. Problem Formulation

Let \( x \in \mathcal{X} = \{ x_i \}_{i=1}^{X} \) be a discrete variable representing a sentence of tokens of length \( T \in \{ 1, \ldots, T_{\text{max}} \} \) from a finite vocabulary \( \mathcal{V} \), where \( T_{\text{max}} \) is the maximum sentence length. The set \( \mathcal{X} \) comprises all sentences we aim to model. The total number of sentences \( X \) is typically unknown and large. Let \( \mathcal{P}(x) \) be the unknown probability of sentence \( x \).

Our goal is to learn a latent variable model given \( N \) fair samples from \( \mathcal{P}(x) \), where \( N \ll X \). To this end, we consider probabilistic auto-encoders, defining distributions over discrete observations \( x \in \mathcal{X} \), and a corresponding continuous latent space, \( z \in \mathbb{R}^d \). They consist of an encoder, \( q_\theta(z|x) \), mapping sentences to a distribution over continuous latent codes, and a corresponding decoder, \( p_\theta(x|z) \), providing a distribution over sentences given a latent code. The joint parameters of the encoder and the decoder are denoted by \( \theta \). Ideally the encoder maps inputs to latent codes from which the decoder can correctly reconstruct the input. We also desire a latent space in which similar sentences (e.g., in structure or content) are mapped to nearby latent codes.

2.1. Encoder-Decoder Specification

In what follows we adapt the architecture proposed by Bowman et al. (2015). Beginning with the generative process, let \( p_\theta^{\text{seq}}(x|z) \) be a conditional auto-regressive distribution over sequences of \( T \) tokens. We express the log probability of a sequence, \( x = (x^1, \ldots, x^T) \), with tokens \( x^k \in \mathcal{V} \) (and a slight abuse of notations), as

\[
\log p_\theta^{\text{seq}}(x|z) = \sum_{k=1}^{T} \log p_\theta^{\text{seq}}(x^k | x^{k-1}, \ldots, x^1, z) \tag{1}
\]

where \( p_\theta^{\text{seq}}(x^k|\cdot) \) is a categorical distribution over \( |\mathcal{V}| \) possible tokens for the \( k^{th} \) token in \( x \), and \( x^0 \equiv \langle \text{SOS} \rangle \) is the start-of-sentence token. According to the model (see Fig. 1), generating a sentence \( x \) with latent code \( z \) entails sampling each token from a distribution conditioned on the latent code

\[
\begin{align*}
\text{the} & \quad \text{GRU} \quad \text{h}_1^1 \quad \text{GRU} \quad \text{h}_2^1 \quad \text{GRU} \quad \text{h}_3^1 \quad \text{GRU} \quad \text{h}_4^1 \quad \text{GRU} \quad \text{h}_5^1 \quad \text{GRU} \\
\langle \text{SOS} \rangle & \quad \text{h}_1^2 \quad \text{GRU} \quad \text{h}_2^2 \quad \text{GRU} \quad \text{h}_3^2 \quad \text{GRU} \quad \text{h}_4^2 \quad \text{GRU} \\
\text{cat} & \quad \text{is} \quad \text{sitting} \quad \langle \text{EOS} \rangle
\end{align*}
\]

Figure 2. The encoder is implemented with GRU. Each word is represented by a parametric embedding. Given the input sequence, the encoder maps the last hidden state to the mean and variance of Gaussian posterior over latent codes, \( q_\theta(z|h_5^n) \).

and previously sampled tokens. Tokens are modelled with a parametric embedding.

The auto-regressive model, \( p_\theta^{\text{seq}}(x|z) \), sums to one over all sequences of a given length. Combining this model with a distribution over sentence lengths \( p(T) \), for \( T \in \{ 1, \ldots, T_{\text{max}} \} \), we obtain the decoder, i.e., a distribution over all sentences in \( \mathcal{X} \):

\[
p_\theta(x|z) = p_\theta^{\text{seq}}(x|z) p(T = \ell). \tag{2}
\]

Here, \( p_\theta(x|z) \) sums to one over all sentences of all lengths. The corresponding marginal \( p_\theta(z) \) is discussed in Sec. 2.3.

The encoder, or posterior distribution over latent codes given a sentence, \( q_\theta(z|x) \), is a conditional distribution over the latent variable \( z \). We take this to be Gaussian whose mean and diagonal covariance are specified by mappings \( \mu_\theta \) and \( \sigma_\theta \):

\[
q_\theta(z|x) = \mathcal{N}(z; \mu_\theta(x), \sigma_\theta(x)) \tag{3}
\]

Linear mappings \( \mu_\theta \) and \( \sigma_\theta \) are computed from the last hidden state of a GRU (Cho et al., 2014) (see Fig. 2).

2.2. Background: MIM Learning Objective

The Mutual Information Machine (MIM), introduced by Livne et al. (2019), is a versatile LVM. Like the VAE, it serves as a framework for representation learning, probability density estimation, and sample generation. Importantly,
MIM learns a model with high mutual information between observations and latent codes, and with robustness against posterior collapse, which has been problematic for VAEs with language data (e.g., Bowman et al. (2015)).

MIM is formulated in terms of several elements. It assumes two anchor distributions, $P(x)$ and $P(z)$, for observations and the latent space, from which one can draw samples. They are fixed and not learned. There is also a parameterized encoder-decoder pair, $q_\theta(z|x)$ and $p_\theta(x|z)$, and parametric marginal distributions $q_\theta(x)$ and $p_\theta(z)$. These parametric elements define joint encoding and decoding distributions:

$$q_\theta(x, z) = q_\theta(z|x) q_\theta(x), \quad (4)$$
$$p_\theta(x, z) = p_\theta(x|z) p_\theta(z). \quad (5)$$

For language modeling we use A-MIM learning, a MIM variant that minimizes a loss defined on the encoding and decoding distributions, with samples drawn from an encoding sample distribution, denoted $M^t_S(x, z)$; i.e.,

$$M^t_S(x, z) = q_\theta(z|x) P(x). \quad (6)$$

The particular loss for A-MIM is a variational upper bound on the joint entropy of the encoding sample distribution, which can be expressed with marginal entropies and mutual information terms. More precisely,

$$\mathcal{L}_{A,MIM}(\theta) = \frac{1}{2} \left( CE \left( M^t_S(x, z), q_\theta(x, z) \right) + CE \left( M^t_S(x, z), p_\theta(x, z) \right) \right) \geq H_{M^t_S}(x) + H_{M^t_S}(z) - I_{M^t_S}(x; z), \quad (7)$$

where $CE(\cdot, \cdot)$ is cross-entropy, $H_{M^t_S}(\cdot)$ is information entropy over distribution $M^t_S$, and $I(\cdot; \cdot)$ is mutual information. Minimizing $\mathcal{L}_{A,MIM}(\theta)$ learns a model with a consistent encoder-decoder, high mutual information, and low marginal entropy (Livne et al., 2019).

### 2.3. Variational Model Marginals

To complete the model specification, we define the model marginals $q_\theta(x)$ and $p_\theta(z)$. To help encourage consistency, and avoid introducing more model parameters, one can define model marginals in terms of marginals of the sample distributions (Bornschein et al., 2015; Livne et al., 2019; Tomczak & Welling, 2017).

We define the model marginal over observations as a marginal over the decoder (Bornschein et al., 2015): i.e.,

$$q_\theta(x) = \mathbb{E}_{P(x)} [p_\theta(x|z)], \quad (8)$$

where the latent anchor is defined to be a standard normal, $P(z) = \mathcal{N}(z; 0, 1)$. Similarly, one can define the model marginal over latent codes as a marginal of the encoder,

$$p_\theta(z) = \mathbb{E}_{P(x)} [q_\theta(z|x)]. \quad (9)$$

The latent marginal is defined as the aggregated posterior, in the spirit of the VampPrior (Tomczak & Welling, 2017).

### 2.4. Tractable Bounds to Loss

Given a training dataset $D = \{x_i\}_{i=1}^N$, an empirical approximation to $\mathcal{L}_{A,MIM}(\theta)$ is

$$\hat{\mathcal{L}}_{A,MIM}(\theta) = -\frac{1}{2N} \sum_{x_i} E_{q_\theta(z|x_i)} \left[ \log q_\theta(z|x_i) q_\theta(x_i) \right]$$

$$-\frac{1}{2N} \sum_{x_i} E_{p_\theta(x|z_i)} \left[ \log p_\theta(x_i|z) p_\theta(z) \right] \quad (10)$$

where $\sum_{x_i}$ denotes $\sum_{x \in D}$, a sum over $N$ fair samples drawn from $P(x)$, as a Monte Carlo approximation to expectation over $P(x)$.

Unfortunately, the empirical loss in Eqn. (10) is intractable since we cannot evaluate the log-probability of the marginals $p_\theta(z)$ and $q_\theta(x)$. In what follows we obtain a tractable empirical bound on the loss in Eqn. (10) for which, with one joint sample, we obtain an unbiased and low-variance estimate of the gradient (i.e., using the reparameterization trick (Kingma & Welling, 2013)).

We first derive a tractable lower bound to $\log q_\theta(x_i) = \log \mathbb{E}_{P(z)} [p_\theta(x_i|z)]$ for any discrete distribution $P(z)$ and any non-negative function $h(x; \cdot) \geq 0$. The inequality in Eqn. (12) follows from log $a \geq \log b$ for $a \geq b$. Using this bound, we express a lower bound to $p_\theta(z)$ as follows,

$$\log \mathbb{E}_{P(x)} [h(x; \cdot)] = \log \sum_i P(x_i) h(x_i; \cdot) \geq \log P(x') h(x'; \cdot), \quad (12)$$

for any sample $x'$, any discrete distribution $P(x)$, and any non-negative function $h(x; \cdot) \geq 0$. The inequality in Eqn. (12) follows from log $a \geq \log b$ for $a \geq b$. Using this bound, we express a lower bound to $p_\theta(z)$ as follows,

$$\log p_\theta(z) = \log \mathbb{E}_{P(x)} [q_\theta(z|x)] \geq \log q_\theta(z|x') + \log P(x') \quad (13)$$

for any sample $x'$. During training, given a joint sample $x_i, z_i \sim q_\theta(z|x) P(x)$, we choose $x' = x_i$. 

---

*SentenceMIM*
We find an unbiased, low variance estimate of the gradient of when the aggregated posterior is a poor fit to the original with implicit priors. With implicit priors, both NLL and As an alternative to the ELBO bound on log likelihood, here we discuss model evaluation under a given empirical target distribution \( T \) and reparameterization. The last term, \( H_P(x) \), is a constant, independent of model parameters and can therefore be ignored during optimization. The resulting learning process is described in Algorithm 1.

To better understand the proposed bounds, we note that MIM achieves good reconstruction by learning posteriors with relatively small variances (i.e., relative to the distance between latent means). Our choice of \( x' = x_i \) exploits this, allowing good gradient estimation, facilitating fast convergence. We further provide empirical evidence for these properties below in Fig. 3.

\[ \hat{L}_{A-MIM} \leq -\frac{1}{N} \sum_i \mathbb{E}_{q_\theta(z|x_i)} \left[ \log p_\theta(x_i|z) \right] \]

Substituting Eqns. (11) and (13) into Eqn. (10) gives the final form of an upper bound on the empirical loss; i.e.,

\[ \Delta \theta \propto -\nabla \hat{L}_{A-MIM}(\theta; D) \quad \text{[Gradient computed through sampling using reparameterization]} \]

We can also view Eqn. (15) as an alternative to the usual ELBO, this bound holds for each data point, independent of model parameters and can therefore be used to change the prior once the encoder and decoder have been trained (e.g., Razavi et al. (2019); van den Oord et al. (2017)). Interestingly, if post-hoc we change the latent prior to be a marginal distribution, as in Eqn. 9, then MELBO can be used to bound the NLL. This is particularly effective when the aggregated posterior is a poor fit to the original Gaussian prior, which is penalized heavily in the EBLO.

Here we discuss model evaluation under a given empirical target distribution \( T(x) \), and in particular, the empirical test set. We start with a bound on \( \log p_\theta(x_i) \), i.e.,

\[ \log p_\theta(x_i) = \log \mathbb{E}_{q_\theta(z|x_i)} \left[ p_\theta(x_i|z) \right] \geq \mathbb{E}_{q_\theta(z|x_i)} \left[ p_\theta(x_i|z) \right] q_\theta(z|x_i) \]

where the second step uses importance sampling, and the variational marginal in Eqn. (9) is defined here under the target empirical distribution \( T(x) \) in the last step. This allows us to choose \( x' = x_i \), motivated by the tendency for MIM to learn highly clustered representations (cf. Fig. 3). We can also view Eqn. (15) as an alternative to the usual ELBO; we refer to it as MELBO (i.e., MIM ELBO). Like the ELBO, this bound holds for each data point, independent of target distribution \( T(x) \).

We can now derive an upper bound on the NLL under \( T \) using the MELBO:

\[ -E_{T(x)} \left[ \log p_\theta(x) \right] \leq -\mathbb{E}_{q_\theta(z|x)} \left( \log p_\theta(x|z) \right) + H_T(x) \]

\[ \leq \frac{1}{N} \sum_{x_i} H_P(x) + \frac{1}{N} \sum_{x_i} \left( \frac{1}{N} \sum_{j=1}^{N_x} \log p_\theta(x_i|z_{i,j}) \right) + \log N \]

3. NLL Evaluation

As an alternative to the ELBO bound on log likelihood, here we propose a new bound that is better suited to models with implicit priors. With implicit priors, both NLL and ELBO are computationally expensive to estimate. Unlike ELBO, the new bound, called MELBO (for MIM-ELBO) does not entail the evaluation of the log-likelihood of the latent model marginal. In the VAE literature, it is common to change the prior once the encoder and decoder have been trained (e.g., Razavi et al. (2019); van den Oord et al. (2017)). Interestingly, if post-hoc we change the latent prior to be a marginal distribution, as in Eqn. 9, then MELBO can be used to bound the NLL. This is particularly effective when the aggregated posterior is a poor fit to the original Gaussian prior, which is penalized heavily in the EBLO.

\[ \text{Learning parameters } \theta \text{ of sentenceMIM} \]

1. while not converged do
2. \( D_{\text{enc}} \leftarrow \{ x, z_{i,j} \sim q_\theta(z|x) \mathcal{P}(x) \}_{j=1}^{N} \)
3. \( \hat{L}_{MIM}(\theta; D) = -\frac{1}{N} \sum_{i=1}^{N} \left( \log p_\theta(x_i|z_i) + \frac{1}{2} \left( \log q_\theta(z_i|x_i) + \log \mathcal{P}(z_i) \right) \right) \)
4. \( \Delta \theta \propto -\nabla \hat{L}_{MIM}(\theta; D) \quad \text{[Gradient computed through sampling using reparameterization]} \)
5. end while

For the sizes of the datasets we consider, MELBO and ELBO tend to be comparable for VAE. Notice that \( \hat{L}_{A-MIM} \) grows with the number of unique sentences in the dataset \( N \).

SentenceMIM

\( \text{1}<\text{EOS}> \), \( <\text{EOS}> \) are a special start/end-of-sentence tokens. The token \(<\text{UNK}>\) represents an out-of-vocabulary word.
draw 100k samples for training, and 10k for validation and testing. For WT103 we draw 200k samples for training, 10k for validation, and retain the original test data. Empty lines and headers were filtered from the WT103 data.

4.2. Architecture and Optimization

Our auto-encoder architecture (Figs. 1 and 2), was adapted from that proposed by Bowman et al. (2015). As is common, we concatenated \( z \) with the input to the decoder (i.e., a “context”, similar to He et al. (2019); Yang et al. (2017); Bowman et al. (2015)). We use the same architecture, parameterization and latent dimensionality for sMIM and a VAE variant called sVAE, for comparison. Training times for sVAE and sMIM are similar.

For PTB we trained models with 1 layer GRU, latent space dimensions of 16D, 128D, and 512D, a 512D hidden state, 300D word embeddings, and 50% embedding dropout. We trained the models with Adam (Kingma & Lei Ba, 2014) with initial learning rate \( lr = 10^{-3} \). The best performing model was trained in less than 30 minutes on a single TITAN Xp 12G GPU. For Yahoo Answers, Yelp15, and WT103 we trained models with 1 layer GRU, latent space dimensions of 32D, 512D, 1024D, a 1024D hidden state, 512D word embeddings, and 50% embedding dropout. We trained these models with SGD (Sutskever et al., 2013), with initial \( lr = 5.0 \), and 0.25 \( L_2 \) gradient clipping.

In all cases we use a learning rate scheduler that scaled the learning rate by 0.25 following two/one epochs (PTB/other datasets, respectively) with no improvement in the validation loss. We used a mini-batch size of 20 in all cases. Following (Sutskever et al., 2014) we feed the input in reverse to the encoder, such that the last hidden state in the encoder depends on the first word of the sentence in the decoder. (This gave slightly better results than with left to right order.)

We trained sVAEs with the regular ELBO, and with KL divergence annealing (denoted “+\( k_l \)”), where a scalar weight on the KL divergence term is increased from 0 to 1 over

| (word level)       | Sentences                                                                 |
|--------------------|---------------------------------------------------------------------------|
|                    | (Tran Validation. Test Vocab. #words (avg.)                              |
| PTB                | 42068 3370 3761 9877 21 ± 10                                           |
| Yahoo             | 100K 10K 10K 37165 76 ± 55                                             |
| Yelp15            | 100K 10K 10K 19730 100 ± 51                                           |
| WikiText-103      | 200K 10K 2185 89247 115 ± 60                                          |
| Everything†       | 442067 33369 33760 105963 94 ± 60                                     |

Table 1. Dataset properties summary for Penn Tree Bank (Marcus et al., 1993), Yahoo Answers and Yelp15 (cf. Yang et al. (2017)), and sampled WikiText-103 (Merity et al., 2016). Everything† is the union of all datasets.

4.3. Language Modelling Results

In what follows we compare the perplexity (PPL) of sMIM, sVAE, other top performing VAEs, and auto-regressive models. For all datasets but PTB, VAE learning with KL annealing was more effective than standard VAE learning; due to the small size of PTB, annealing produced over-fitting. We remove the \(<EOS>\) token during evaluation, allowing fair PPL comparison with auto-regressive models\(^2\).

10k mini-batches to lower the risk of posterior collapse and improve the learned models (Bowman et al., 2015). We use no loss manipulation heuristics in the optimization of sMIM.

In what follows we compare the perplexity (PPL) of sMIM, sVAE, other top performing VAEs, and auto-regressive models. For all datasets but PTB, VAE learning with KL annealing was more effective than standard VAE learning; due to the small size of PTB, annealing produced over-fitting. We remove the \(<EOS>\) token during evaluation, allowing fair PPL comparison with auto-regressive models\(^2\).

Tables 2-5 show results for PTB, Yelp15, Yahoo Answers, and WT103. Model sMIM (1024)\(^†\) is trained on all datasets (i.e., PTB, Yahoo Answers, Yelp15 and WT103). The BLEU-1 score is computed between test sentences and their reconstructions (higher is better). PPL and NLL (lower is better) are bounded with MELBO (Eqn. (16)). PPL comparison with auto-regressive models\(^2\).

| Model             | PPL (std) | NLL (KLD) | BLEU | \( \theta \) |
|-------------------|-----------|-----------|------|-------------|
| sVAE (16)         | \( \leq 110.72 \) | \( \leq 106.84 \) | 0.124 | 11M         |
| sVAE (128)        | \( \leq 148.18 \) | \( \leq 113.46 \) | 0.25  | 11M         |
| sVAE (512)        | \( \leq 158.31 \) | \( \leq 114.96 \) | 0.118 | 11M         |
| sMIM (16)         | \( \leq 121.44 \) | \( \leq 108.93 \) | 0.116 | 12M         |
| sMIM (128)        | \( \leq 113.3 \) | \( \leq 116.76 \) | 0.118 | 12M         |
| sMIM (512)        | \( \leq 19.53 \) | \( \leq 67.46 \) | 0.118 | 12M         |
| sMIM (1024)       | \( \leq 4.6 \) | \( \leq 24.66 \) | 0.118 | 12M         |
| auto-regressive   |            |            |      |             |
| (a) VAE-LSTM (13) | 119       | 101 [2]   |      |             |
| (b) VAEAE (32)    | \( \leq 53.44 \) | \( \leq 87.2 [12.51] \) | 0.118 | 12M         |
| (c) HR-VAE (256)  | 43        | 79 [10.4] |      |             |
| (d) GPT-2 full†   | 35.76     | 1542M     |      |             |

Table 2. PPL and NLL results for PTB bounded with MELBO , averaged over 10 runs (see text for details). PPL\(^*\) and NLL\(^*\) are bounded with ELBO. Models\(^†\) use extra training data. (a) Bowman et al. (2015); (b) Le Fang (2019); (c) Li et al. (2019a); (d) Radford et al. (2019). A test set with 2 \( \times 10^9 \) samples is required for MELBO of best performing sMIM to reach the PPL of next best model. (a)\(^†\) inconsistencies in PPL values can be explained by Bowman et al. (2015) including \(<EOS>\) during evaluation.
Table 3. PPL and NLL results for Yelp15 bounded with MELBO, averaged over 10 runs (see text for details). PPL * and NLL † are bounded with ELBO. Models ‡ use extra training data. (a) Yang et al. (2017); (b) Kim et al. (2018); (c-e) He et al. (2019); (f) Guu et al. (2017). Results in rows (a–d) are taken from He et al. (2019). A test set with $7 \times 10^{44}$ samples is required for MELBO of best performing sMIM to reach the PPL of next best model.

Table 4. PPL and NLL results for Yahoo Answers bounded with MELBO, averaged over 10 runs (see text for details). PPL * and NLL † are bounded with ELBO. Models ‡ use extra training data. (a) Yang et al. (2017); (b) Kim et al. (2018); (c-e) He et al. (2019); (f) Guu et al. (2017). Results in rows (a–e) are taken from He et al. (2019). A test set with $7.2 \times 10^{44}$ samples is required for MELBO of best performing sMIM to reach the PPL of next best model.

Table 5. PPL and NLL results for WT103 bounded with MELBO, averaged over 10 runs (see text for details). PPL * and NLL † are bounded with ELBO. Models ‡ use extra training data. (a) Shoeybi et al. (2019); (b) Krause et al. (2019); (c) Radford et al. (2019).
Figure 3. Histograms of log probabilities of test data for sMIM and sVAE trained on PTB: Overlap between curves indicates potential for poor reconstruction of input sentences. (a) Histograms of $\log p_{\theta}(x_i | z_j)$ for $z_j \sim q_{\theta}(z | x_i)$ when $i = j$ (same input), and when $i \neq j$ (when $x_i$ is evaluated with the decoder distribution from a latent code associated with a different input sentence). (b) Histograms of $\log q_{\theta}(z | x_i)$ for $z_i \sim q_{\theta}(z | x_i)$, when conditioned on the same input $i = j$, or a different input $i \neq j$.

Table 6. Entropy of the latent/hIDDEN distribution for sMIM and AE (estimated using NN entropy estimator (Kraskov et al., 2004)). (S, M) latent dimensions correspond to (16D, 128D) in PTB and (32D, 512D) in Yelp15 and Yahoo Answers. For comparison, columns $N(d)$ gives the entropy of a standard Normal in $\mathbb{R}^d$.

| Data     | sMIM (S) $N(S)$ | sMIM (M) $N(M)$ | sMIM (L) $N(L)$ | AE (S) $N(S)$ | AE (M) $N(M)$ |
|----------|----------------|----------------|----------------|--------------|--------------|
| PTB      | 11.54          | 22.7           | 35.95          | 53.73        | 181.62       | 259.34       |
| Yelp15   | 32.22          | 45.4           | 73.03          | 186.18       | 726.49       | 917.0        |
| Yahoo    | 23.61          | 45.4           | 76.21          | 155.47       | 726.49       | 1003.26      |

Table 7. BLEU results for reconstruction of sMIM and AE. sMIM demonstrates empirical robustness to over-fitting, when compared to AE. (S, M, L) latent dimensions correspond to (16D, 128D, 512D) in PTB and (32D, 512D, 1024D) in Yelp15 and Yahoo Answers.

| Model                                      | P@1 | MRR |
|--------------------------------------------|-----|-----|
| AP-CNN (dos Santos et al., 2016)           | 0.560 | 0.726 |
| AP-BiLSTM (dos Santos et al., 2016)        | 0.568 | 0.731 |
| HyperQA (Tay et al., 2017a)                | 0.683 | 0.801 |
| sMIM (512)                                 | 0.683 | 0.818 |
| AE (512)                                   | 0.58  | 0.814 |
| sMIM (1024)                                | 0.753 | 0.861 |

Table 8. YahooCQA results for sMIM, AE, and single-task models (higher is better). Results are averaged over 10 runs (stddev < 0.002). sMIM (1024) is pre-trained on PTB, Yahoo Answers, Yelp15 and WT103. P@1 and MRR are defined in Sec. 4.6.

Compressed and clustered representation).

We show the empirical entropy (estimated using NN entropy estimator (Kraskov et al., 2004)) of the hidden/latent codes in Table 6. It is clear that sMIM learns a low entropy representation (i.e., lower than the anchor $P(z)$), whereas AE has no notion of a latent distribution, leading to high information entropy in the hidden state (i.e., more uniformly distributed, with less structure).

Table 7 shows BLEU values for sMIM and AE with the same architecture. Interestingly, the added latent stochasticity in sMIM helps mitigate over-fitting, while AE is more sensitive to the choice of architecture (i.e., stronger model might over-fit), as evident for Yahoo Answers. In addition, learning a latent distribution makes sMIM a useful model for downstream tasks, as we discuss next.

4.6. Question-Answering

To demonstrate the versatility of sMIM, we consider a downstream task in which sMIM (512) is pre-trained on Yahoo Answers, then used for question-answering on YahooCQA (Tay et al., 2017b), with no fine-tuning. The YahooCQA vocabulary has 116,900 tokens, with training, validation and test sets having 253K, 31.7K and 31.7K QA pairs, respectively. These sets were constructed by taking a subset of the QA pairs from Yahoo Answers, from which each question is then paired with another 2-4 answers (not from Yahoo Answers). Thus each question, of 5-50 tokens, has 3-5 possible ranked answers (1 is best). Let $Q_i$ denote the $i$th question, and let $\{A^k_i\}_{k=1}^{K_i}$ be the $K_i$ corresponding answers, ordered such that $A^k_i$ has rank $k$. To match the format of QA pairs

sentence are extremely unlikely to be generated from sMIM. This is not the case for sVAE, where the histograms overlap. In other words, sMIM effectively maps sentences to non-overlapping regions of the latent space, allowing good reconstruction. By comparison, with sVAE sentences are mapped to overlapping regions of the latent space, which hinders accurate reconstruction.

4.5. Comparison of sMIM to Auto-encoders

To provide additional insight into the latent representation learned by sMIM, we contrast sMIM with a deterministic sequence-auto-encoder (AE) of the same architecture. We train AEs by keeping the reconstruction term in the sVAE loss, discarding the KL divergence term, and taking the mean of the posterior to be the hidden state that is fed to the decoder (i.e., $z_i = \mathbb{E}_{z'(x_i)} [q_{\theta}(z' | x_i)]$). While AEs maximize the reconstruction between the observations and the hidden state, they do not learn a distribution over latent codes. MIM, on the other hand, learns a low entropy distribution (i.e., a
The company didn’t break out its fourth-quarter results.

Table 9. Sampled answers from Yahoo Answers sMIM (1024).

| Question | Answer |
|----------|--------|
| the company didn’t break out its fourth-quarter results | Yahoo Answers | sMIM (1024) |
| A: you can find out the county jail, or call your local police station. | eBay | Yelp15 |
| A: the most notable is in the Netherlands. | Ask yourself! | Yelp15 |
| A: how do you list the history in the search field? | eBay | WT103 |
| A: what is the best question to ask? | Ask yourself! | WT103 |
| A: need to find somewhere to save baseball cards. | eBay | Yelp15 |
| A: what’s the opposite of opposite? | I thought it really helps. | Yelp15 |

Table 10. Interpolation results between latent codes of input sentences (with gray) from Yelp15 for sMIM (1024).

| (D) | <SOS> the company didn’t break out its fourth-quarter results |
| (M) | the company didn’t break out its results |
| (R) | the company didn’t break out its fourth-quarter results |
| (P) | the company didn’t accurately break out its results |

Table 11. Reconstruction results for sMIM (512) model trained on PTB. We denote: (D) Data sample; (M) Mean (latent) reconstruction; (R) Reconstruction; (P) Perturbed (latent) reconstruction.

| SentenceMIM | <SOS> awesome food, just awesome | top notch beer selection, great staff, beer garden is great setting. |
|-------------|---------------------------------|---------------------------------------------------------------------|
| (D)         | awesome food, just awesome | top notch beer selection, great staff, beer garden is great setting. |
| (M)         | awesome food, just awesome | top notch beer selection, great staff, beer garden is great setting. |
| (R)         | awesome food, just awesome | top notch beer selection, great staff, beer garden is great setting. |
| (P)         | awesome food, just awesome | top notch beer selection, great staff, beer garden is great setting. |

4.7. Reconstruction, Interpolation, and Perturbation

As a final exploration of sMIM, we probe the learned representation, demonstrating that sMIM learns a dense, meaningful latent space. We present latent interpolation results in Table 10 for samples (i.e., reviews) with the different ratings from Yelp5. Interpolation entails sampling \( x \sim p_\theta(x|z_a) \) where \( z_a \) is equispaced at equispaces points between two latent codes, \( z_i \sim q_\theta(z|x_i) \), and \( z_j \sim q_\theta(z|x_j) \).

Next we show reconstruction, and perturbation results for for sMIM (512) trained on PTB. Figure 11 shows four sentences: (D) the input sentence; (M) the mean reconstruction given the posterior mean \( z \); (R) a reconstruction given a random sample \( z \) from the posterior; and (P) a perturbed reconstruction, given a sample \( z \) from a Gaussian distribution with 10 times the posterior standard deviation. The high mutual information learned by sMIM leads to good reconstruction, as clear in (M) and (R). sMIM also demonstrates good clustering in the latent space, shown here by the great similarity of (R) and (P).

5. Conclusions

This paper introduces a new generative auto-encoder for language modeling, trained with A-MIM learning. The resulting framework learns an encoder that provides a continuous distribution over latent codes for a sentence, from which one can reconstruct, generate and interpolate sentences. In particular, compared to recent attempts to use VAEs for language learning, A-MIM provides models with high mutual information between observations and latent codes, im-
SentenceMIM

proved reconstruction, and it avoids posterior collapse. On PTB, Yaoo Anwers, and Yelp15 we obtain state-of-the-art perplexity results, with competitive results on Wiki103. We also use the latent representation for a downstream question-answering task on YahooCQA with state-of-the-art results. Finally, we demonstrate language generation, perturbation and interpolation using the latent representation. To the best of our knowledge, this is the first LVM for text that achieves competitive performance with non-LVM models.

References

Alemi, A. A., Poole, B., Fischer, I., Dillon, J. V., Saurous, R. A., and Murphy, K. An information-theoretic analysis of deep latent-variable models. CoRR, abs/1711.00464, 2017.

Bird, S. NLTK: The natural language toolkit. ArXiv, cs.CL/0205028, 2002.

Bornschein, J., Shabanian, S., Fischer, A., and Bengio, Y. Bidirectional Helmholtz machines. CoRR, abs/1506.03877, 2015.

Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A. M., Józefowicz, R., and Bengio, S. Generating sentences from a continuous space. CoRR, abs/1511.06349, 2015.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv, 2014.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using real nvp. ICLR, 2017.

dos Santos, C. N., Tan, M., Xiang, B., and Zhou, B. Attentive pooling networks. CoRR, abs/1602.03609, 2016.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In Advances in neural information processing systems, pp. 2672–2680, 2014.

Guu, K., Hashimoto, T. B., Oren, Y., and Liang, P. Generating sentences by editing prototypes. ACL, 6:437–450, 2017.

He, J., Spokony, D., Neubig, G., and Berg-Kirkpatrick, T. Lagging inference networks and posterior collapse in variational autoencoders. In ICLR, 2019.

Kim, Y., Wiseman, S., Miller, A., Sontag, D., and Rush, A. Semi-amortized variational autoencoders. In Dy, J. and Krause, A. (eds.), ICML, volume 80 of Proceedings of Machine Learning Research, pp. 2678–2687, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/kim18e.html.

Kingma, D. P. and Lei Ba, J. ADAM: A method for stochastic optimization. In ICLR, 2014.

Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. In ICLR, 2013.

Krause, A., Stögbauer, H., and Grassberger, P. Estimating mutual information. Phys. Rev. E, 69: 066138, Jun 2004. doi: 10.1103/PhysRevE.69.066138. URL https://link.aps.org/doi/10.1103/PhysRevE.69.066138.

Krause, B., Kahembwe, E., Murray, I., and Renals, S. Dynamic evaluation of transformer language models. CoRR, abs/1904.08378, 2019. URL http://arxiv.org/abs/1904.08378.

Kruengkrai, C. Better exploiting latent variables in text modeling. ACL, pp. 5527–5532, 2019.

Le Fang, Chunyuan Li, J. G. W. D. C. C. Implicit deep latent variable models for text generation. In EMNLP, 2019.

Li, R., Li, X., Lin, C., Collinson, M., and Mao, R. A stable variational autoencoder for text modelling. In INLG, 2019a.

Li, R., Li, X., Lin, C., Collinson, M., and Mao, R. A stable variational autoencoder for text modelling. INLG, pp. 594–599, 2019b.

Livne, M., Swersky, K., and Fleet, D. J. MIM: Mutual Information Machine. arXiv e-prints, 2019.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. Building a large annotated corpus of English: The Penn Treebank. Comput. Linguist., 19(2):313–330, June 1993. ISSN 0891-2017. URL http://dl.acm.org/citation.cfm?id=972470.972475.

Merity, S., Xiong, C., Bradbury, J., and Socher, R. Pointer sentinel mixture models. CoRR, abs/1609.07843, 2016. URL http://arxiv.org/abs/1609.07843.

Merity, S., Keskar, N. S., and Socher, R. Regularizing and optimizing LSTM language models. CoRR, abs/1708.02182, 2017. URL http://arxiv.org/abs/1708.02182.

Oord, A. v. d., Kalchbrenner, N., and Kavukcuoglu, K. Pixel recurrent neural networks. International Conference on Machine Learning, 2016.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. BLEU: a method for automatic evaluation of machine translation. In ACL, 2001.
Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners. 2019.

Rae, J. W., Dyer, C., Dayan, P., and Lillicrap, T. P. Fast parametric learning with activation memorization. CoRR, abs/1803.10049, 2018. URL http://arxiv.org/abs/1803.10049.

Razavi, A., van den Oord, A., Poole, B., and Vinyals, O. Preventing posterior collapse with delta-VAEs. CoRR, abs/1901.03416, 2019.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep tive Models. In ICML, 2014.

Shoeybi, M., Patwary, M., Puri, R., LeGresley, P., Casper, J., and Catanzaro, B. Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism. arXiv e-prints, art. arXiv:1909.08053, Sep 2019.

Sutskever, I., Martens, J., Dahl, G., and Hinton, G. On the importance of initialization and momentum in deep learning. In Dasgupta, S. and McAllester, D. (eds.), ICML, volume 28 of Proceedings of Machine Learning Research, pp. 1139–1147, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR.

Sutskever, I., Vinyals, O., and Le, Q. V. Sequence to sequence learning with neural networks. In NIPS, NIPS, pp. 3104–3112, Cambridge, MA, USA, 2014. MIT Press. URL http://dl.acm.org/citation.cfm?id=2969033.2969173.

Tay, Y., Luu, A. T., and Hui, S. C. Enabling efficient question answer retrieval via hyperbolic neural networks. CoRR, abs/1707.07847, 2017a. URL http://arxiv.org/abs/1707.07847.

Tay, Y., Phan, M. C., Luu, A. T., and Hui, S. C. Learning to rank question answer pairs with holographic dual LSTM architecture. In SIGIR, pp. 695–704, 2017b. doi: 10.1145/3077136.3080790. URL http://doi.acm.org/10.1145/3077136.3080790.

Tomczak, J. M. and Welling, M. VAE with a vampprior. CoRR, abs/1705.07120, 2017. URL http://arxiv.org/abs/1705.07120.

van den Oord, A., Vinyals, O., and Kavukcuoglu, K. Neural discrete representation learning. In NIPS, pp. 6306–6315, 2017.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In NIPS, pp. 5998–6008, 2017.

Wang, D., Gong, C., and Liu, Q. Improving neural language modeling via adversarial training. In Chaudhuri, K. and Salakhutdinov, R. (eds.), ICML, volume 97 of Proceedings of Machine Learning Research, pp. 6555–6565, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL http://proceedings.mlr.press/v97/wang19f.html.

Yang, Z., Hu, Z., Salakhutdinov, R., and Berg-Kirkpatrick, T. Improved variational autoencoders for text modeling using dilated convolutions. CoRR, abs/1702.08139, 2017. URL http://arxiv.org/abs/1702.08139.

Zhao, S., Song, J., and Ermon, S. A Lagrangian perspective on latent variable generative models. UAI, Jul 2018.
A. Distribution of Sentence Lengths

![Histograms of sentence lengths per dataset](image)

*Figure 4.* Here we present histograms of sentence lengths per dataset. The dashed line is the average sentence length.

Fig. 4 shows histograms of sentence lengths. Notice that PTB sentences are significantly shorter than other datasets. As a result, sMIM is somewhat better able to learn a representation that is well suited for reconstruction. Other datasets, with longer sentences, are more challenging, especially with the simple architecture used here (*i.e.*, 1 later GRU). We believe that implementing sMIM with an architecture that better handles long-term dependencies (*e.g.*, transformers) might help.

B. Effect of Sample Set Size on MELBO

![Plots showing effect of sample set size](image)

*Figure 5.* Plots show the effect of sample set size (*K* on x axis) on NLL computed with MELBO upper bound (y axis). For each plot we draw *K* random samples from a test set of size *N*, and compute the bound (*i.e.*, denoted a trial). We repeat the trial *max*(*N* − *K*, 500) times, and compute the mean NLL and the standard deviation. The solid curve depicts the mean NLL; blue shade is 1 standard deviation (over multiple trials). The dashed line is the extrapolated NLL (*i.e.*, see text for details). Red cross marks are NLL values of best performing sMIM model (left mark, for *K* = *N*), and best performing non-sMIM model (right mark). We note that for *K* ⪆ 10^3 the variance in all cases cannot account for the NLL gap.

Here we consider how MELBO, as a bound on NLL, depends on number of test samples. Our goal is to empirically show that bounding NLL with MELBO is robust to the test set used, and that the bound has a reasonably low variance. Fig. 5 shows the dependence of MELBO on the size of a test sample set. For each value of *K*, up to the full test set size, *N*, we randomly sample *K* points in each of several trials, and then plot the mean and standard deviation of the MELBO bound over trials. The solid line shows the mean NLL, as a function of *K*, the standard deviation of which is shown in blue. Once *K* is 1000 or more, the standard deviation is very small, indicating that the specific test sample does not have a significant effect on the bound. In particular, at that point the standard deviation is less than 3.1% of the MELBO bound.

The dashed curve is the extrapolated NLL bound, assuming the average reconstruction error remain constant. Red crosses indicate the MELBO bounds for the full test set (*K* = *N*) and for a test set sufficiently large that the bound equals the NLL of the best performing non-sMIM model; the required sample sizes are orders of magnitude above *N*. This also indicates that the sizes of existing test sets do not account for the large gap in perplexity between sMIM and other models.

C. Comparison of NLL in MIM and VAE

Figures 6-8 depict histograms of ELBO/MELBO values for sentences, for sVAE and sMIM with different latent dimensions. While a less expressive sMIM behaves much like sVAE, the difference is clearer as the expressiveness of the model increases. Here, sVAE does not appear to effectively use the increased expressiveness for better modelling. We hypothesize that the added sVAE expressiveness is used to better match the posterior to the prior, resulting in posterior collapse. sMIM uses the increased expressiveness to increase mutual information.
Figure 6. Histograms of MELBO (sMIM) and ELBO (sVAE) values versus latent dimension for PTB. Dashed black line is the mean.

Figure 7. Histograms of MELBO (sMIM) and ELBO (sVAE) values versus latent dimension for Yelp15. Dashed black line is the mean.

Figure 8. Histograms of MELBO (sMIM) and ELBO (sVAE) versus latent dimension for Yahoo Answers.
D. Additional Results

D.1. Reconstruction

| Model         | Sentence                                                                 |
|---------------|---------------------------------------------------------------------------|
| sMIM (512)    | <SOS> there was no panic                                                 |
|               | (D) there was no panic <EOS>                                             |
|               | (M) there was no panic <EOS>                                             |
|               | (R) there was no panic <EOS>                                             |
|               | (P) there was no panic <EOS>                                             |
|               | (AE) there was no panic <EOS>                                            |
| sMIM (1024) † | <SOS> there was no panic                                                 |
|               | (D) there was no panic <EOS>                                             |
|               | (M) there was no panic <EOS>                                             |
|               | (R) there was no panic <EOS>                                             |
|               | (P) there was no panic <EOS>                                             |
|               | (AE) there was no panic <EOS>                                            |
|               | <SOS> the company didn't break out its fourth-quarter results             |
|               | (D) the company didn't break out its fourth-quarter results <EOS>        |
|               | (M) the company didn't break out its results <EOS>                       |
|               | (R) the company didn't break out its results <EOS>                       |
|               | (P) the company didn't break out its results <EOS>                       |
|               | (AE) the company didn't break out results <EOS>                          |
|               | <SOS> it had planned a strike vote for next sunday but that has been pushed back indefinitely |
|               | (D) it had planned a strike vote for next sunday but that has been pushed back indefinitely |
|               | (M) it had a weakening for promotional planned but that has pushed aside back but so far away <EOS> |
|               | (R) it had planned a strike planned for next week but that continues has been pushed back pushed <EOS> |
|               | (P) it had a strike with state west airline but so that it has slashed its spending but so far said he would be subject by far <EOS> |
|               | (AE) it had planned a strike vote for next sunday but that has been pushed back indefinitely |

Table 12. Reconstruction results for models trained on PTB. We denote: (D) Data sample; (M) Mean (latent) reconstruction; (R) Reconstruction; (P) Perturbed (latent) reconstruction; (AE) Reconstruction of AE.

Here we provide reconstruction results for PTB (Fig. 12), Yelp15 (Fig. 13), and Yahoo Answers (Fig. 14). Each figure shows (D) Data sample; (M) Mean (latent) reconstruction (i.e., $z_i = \mathbb{E} [q_\theta (z | x_i)]$); (R) Reconstruction (i.e., $z_i \sim q_\theta (z | x_i)$); (P) Perturbed (latent) reconstruction (i.e., $z_i \sim q_\theta (z | x_i; \mu_i, 10\sigma_i)$); (AE) Reconstruction of AE. We compare the best performing sMIM model to an AE with the same architecture, and to sMIM (1024) † (i.e., the model trained on the Everything dataset).

Interestingly, AEs tend to perform worse for longer sentences, when compared to sMIM. We attribute this to the higher latent entropy, which leads to non-semantic errors (i.e., nearby latent codes are less similar compared to MIM). Another interesting point is how the reconstruction (R), is better in many cases than the reconstruction given the mean latent code from the encoder (M) (i.e., which have the highest probability density). We attribute that to the fact that most probability mass in a high dimensional Gaussian in $d >> 1$ dimensional space and $\sigma$ standard deviation is concentrated in around a sphere of radius $r \approx \sigma \sqrt{d}$. As a result the probability mass around the mean is low, and sampling from the mean is less likely to represent the input sentence $x_i$. This also explains how perturbations of up to 10 standard deviations might result in good reconstructions. Finally, we point how sMIM (1024) †, trained on Everything, does a better job handling longer sentences.
Table 13. Reconstruction results for models trained on Yelp15. We denote: (D) Data sample; (M) Mean (latent) reconstruction; (R) Reconstruction; (P) Perturbed (latent) reconstruction; (AE) Reconstruction of AE.

| Model       | SentenceMIM | sMIM (1024) | sMIM (1024) |
|-------------|-------------|-------------|-------------|
| (3 stars)   | <EOS>       | decent price. fast. ok staff. ... but it is fast food so i can't rate any higher than 3. | | |
| (M)         | decent       | superior. Decent staff. But it is fast food so it is not so fast. I can't rate any food 3. | | |
| (K)         | decent price | superior. Decent staff. But it is fast food so it is not so fast. I can't rate any food 3. | | |
| (P)         | decent price | fast. Ok. Fast food. But it is ok. So I can't rate any higher than fast. Food is marginal. | | |
| (AE)        | decent price | fast. Ok. Fast food. But it is ok. So I can't rate any higher than fast. | | |
| (4 stars)   | <EOS>       | excellent wings. Great service. 100% smoked wings, great flavor. I will definitely be back. Okra is great too. | | |
| (M)         | excellent wings | great service. 100% smoked wings. Great flavor. Great fries. Definitely will be back. I had too big fat. | | |
| (K)         | excellent wings | great service. Great wings. 100% superior. Great flavor. Great fries. Definitely will be back. | | |
| (P)         | excellent wings | great service. Great wings. 100% superior. Great flavor. Great fries. Definitely will be back. | | |
| (AE)        | excellent wings | great service. Great wings. 100% superior. Great flavor. Great fries. Definitely will be back. | | |
| (5 stars)   | <EOS>       | delicious! The sandwiches are really good and the meat is top quality. It's also nice grabbing an exotic item from the shelf for dessert. | | |
| (M)         | delicious! | The meat really are good and the quality is nice. It's also tempting for notches lovers from the roasters an item top. | | |
| (K)         | delicious! | The sandwiches are really good and the meat is quality. It's also nice for shipping from the top floor an unhygienic machine. | | |
| (P)         | delicious! | The sandwiches are really good and the meat is quality. It's also nice for shipping from the top floor an unhygienic machine. | | |
| (AE)        | delicious! | The sandwiches are really good and the quality is top notch. It's also an item for meat quality memories. | | |

Table 14. Reconstruction results for models trained on Yahoo Answers. We denote: (D) Data sample; (M) Mean (latent) reconstruction; (R) Reconstruction; (P) Perturbed (latent) reconstruction; (AE) Reconstruction of AE.

| Model       | SentenceMIM | sMIM (1024) | sMIM (1024) |
|-------------|-------------|-------------|-------------|
| (Sports)    | <EOS>       | are you regular or googy? Regularly goofy | | |
| (M)         | are you regular or googy? Regularly gritty | are you regular or googy? Regularly gritty | | |
| (K)         | are you regular or googy? Regularly gritty | are you regular or googy? Regularly gritty | | |
| (P)         | are you regular or googy? Regularly gritty | are you regular or googy? Regularly gritty | | |
| (AE)        | are you regular or googy? Regularly gritty | are you regular or googy? Regularly gritty | | |
| (Health)    | <EOS>       | how do you start to like yourself? I was taught by my parents. | | |
| (M)         | how do you start to like yourself? | I would like to meet my parents by | | |
| (K)         | how do you start to like yourself? | I was taught by my parents. | | |
| (P)         | how do you start to like yourself? | I was taught by my parents. | | |
| (AE)        | how do you start to like yourself? | I was taught by my parents. | | |
| (Business & Finance) | <EOS> | how can i find someone in spain? I'm in spain today, what do you want? | | |
| (M)         | how can i find someone in spain? | I'm in spain today, what do you want? | | |
| (K)         | how can i find someone in spain? | I'm in spain today, what do you want? | | |
| (P)         | how can i find someone in spain? | I'm in spain today, what do you want? | | |
| (AE)        | how can i find someone in spain? | I'm in spain today, what do you want? | | |
### D.2. Interpolation

| sMIM (512)                                                                 | sMIM (1024) †                                                                 |
|----------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| <SOS> thanks to modern medicine more couples are growing old together      | <SOS> thanks to modern medicine more modern couples are growing together than  |
| ● to growing small businesses are growing more rapidly growing <EOS>        | <EOS>                                                                         |
| ● growing to more areas are growing preventing black trends <EOS>          | ● thanks to modern medicine more growing peaceful couples form <EOS>          |
| ● growing to the growing industry are growing more rapidly growing than <EOS> | ● thanks to medicine rosen modern more are growing together governing <EOS>   |
| ● growing to the exact industry has been growing more sophisticated six months <EOS> | ● thanks to modern cancer more are growing peaceful couples form <EOS>        |
| ● politics the growing issue are not to mention closely although other prospective products <EOS> | ● thanks to modern medicine rosen modern more are growing together governing <EOS> |
| ● the system is growing enough to make not radical an article <EOS>        | ● the system is growing enough to make not radical an article <EOS>           |
| ● the system is reducing compliance not to consider an article <EOS>       | ● the system is reducing compliance not to consider an article <EOS>          |
| ● the system is the problem system not an effective <EOS>                 | ● the system is the problem system not a church system <EOS>                 |
| ● the system is the system not knowing an individual <EOS>                | ● the system is the system not knowing an individual <EOS>                   |
| ● the system is the system not an encouraging problem <EOS>               | ● the system is the system not knowing an individual <EOS>                   |
| ● the system is the system not an investment fund <EOS>                   | ● the system is the system not an investment fund <EOS>                     |
| ● the system is the problem not an office <EOS>                          | ● the system is the problem not an individual member <EOS>                   |
| ● the system is not the problem for an individual <EOS>                  | ● the system is not the problem for an individual member <EOS>              |
| ● the system is not the problem not an effective <EOS>                   | ● the system is not the individual problem member can <EOS>                 |
| ● the system is not knowing the system not an individual <EOS>            | ● the system is not the individual problem an can <EOS>                     |
| ● the system is not encouraging the system not an individual <EOS>        | ● the system is not the individual problem an individual member <EOS>        |
| ● xtra the system is not even critical <EOS>                            | ● the system is not the individual problem an individual member <EOS>        |
| ● sony denies the declines to secure <EOS>                              | ● sony denies the declines to secure <EOS>                                  |
| ● everyone brought the stock to comment <EOS>                           | ● sony denies the declines to secure <EOS>                                  |
| ● kellogg declines to comment <EOS>                                     | ● kellogg declines to comment <EOS>                                         |
| ● <SOS> sony itself declines to comment                                   | ● <SOS> sony itself declines to comment                                     |

*Table 15. Interpolation results between latent codes of input sentences (with gray) from PTB.*

Here we provide interpolation results for PTB (Fig. 15), Yelp15 (Fig. 16), and Yahoo Answers (Fig. 17). We compare the best performing sMIM model to sMIM (1024) †. Interestingly, both models appear to have learned a dense latent space, with sMIM (1024) † roughly staying within the domain of each dataset. This is surprising since the latent space of sMIM (1024) † jointly represents all datasets.
Table 16. Interpolation results between latent codes of input sentences (with gray) from Yelp15.

| SentenceMIM (1024) | SentenceMIM (1024) |
|--------------------|--------------------|
| • as in china phoenix - this is one of the better ones . get there early - they fill up fast ! <EOS> | • as in phoenix goes this is - better than one of the newest ones . get there early - they fill up fast ! <EOS> |
| • as far in san jose - this is one of the better ones . fast get up early ! there they fill up fast for u ! <EOS> | • as shore goes in phoenix - this is one of the better bbq . fast ! they get up there early - men dinner . <EOS> |
| • as pei wei goes in this phoenix - - one of the best ones . get there early ! they picked up fast food items is better . <EOS> | • as dean goes in phoenix this is the list of bbq . - one not goes fast - get there early ! they fill up fast . <EOS> |
| • oxtail yo buffet in pittsburgh as the owners goes - better . this is not one of those fast food places . fill up there get the hot ! <EOS> | • veal as rocks as this goes in the phoenix area . - one of food is not better quick enough they get . 2 enchiladas up ! <EOS> |
| • ah circle k ! not as bad in the food . thankfully - this one is one of the best bbq joints here ! service was fast friendly . <EOS> | • kohrs as molasses as comparing goes in the food . not sure is one of this better ones - the only ones for fat . thumbs squeeze there ! <EOS> |
| • ehh = cider as the food goes . not bad for service ! - in many fast the only ones available is this . you can get better steak anywhere else ! <EOS> | • omg = rainbow not as the food goes . congrats service ! this is one of the hot spots for only frozen hot - you can . eat on carts there . <EOS> |
| • min spartelz food not the best . wicked spoon ! service is brutal only fast for the hot mexican lv . everything else on this planet as can you get . <EOS> | • = frozen food ! not the best . only frozen hot as for you shall pick the ice cream - loved everything else on wednesday ! <EOS> |
| • frankie food not soo the best . service = horrible ! only drawback frozen for these hike . everything you can pass on the juke planet . <EOS> | • = food not only the best . frozen service ! everything else for the frozen yogurt company . absolute hot tea during normal on as they can . <EOS> |
| • food not the best service . knocking only 99 cents ! for the hot buffet everything . | • = food not . the best frozen service ! only five stars for the water suppose . hot things you can smell on budget . <EOS> |
| beef & broccoli on the sip polo you can pass . <EOS> | • gelato food ! not sure the best . frozen seared only wish you can mix for the frozen hot chocolate frozen . service only best for you ate here twice although the frozen yogurt . delicious atmosphere on everything else . <EOS> |
| • food not the best . service = horrible ! only flopped for the paella everything & rum . you can find everything on the strip . <EOS> | • = food = not the best . frozen service ! only for five stars during the san francisco frozen hot service ! everything else explains . <EOS> |
| • 2 words ! not amazing food . the best <unk> music and service they had can earned a better meal at xx . everything else on bill for me . <EOS> | • = food = not the best . frozen hot service ! only website for the frozen hot chocolate . you can grab everything else on . <EOS> |
| • snottsdale act ! ! no mia <unk> at the food and wished you not a fan . delicious lunch & dessert better choices for desert but they had blackjack . <EOS> | • food = not the best . frozen service ! only for five stars during the san francisco frozen chicken . everything else on could not give thumbs . <EOS> |
| • husbands cher ! wish they had <unk> dessert at the bellagio and not a great lunch selection . food better tasting wise but sadly serves and dessert selection . <EOS> | • food = not ! the frozen yogurt . service only best for you ate here twice although the frozen yogurt . delicious atmosphere on everything else . <EOS> |
| • sooooo ! pretzel panera <unk> they had at a better selection and the food sucked but nothing memorable a dessert . surely great value and better mayonnaise desserts . <EOS> | • gelato food ! not sure the best . frozen seared only wish you can mix for the frozen hot chocolate frozen . service on and everything else explains . <EOS> |
| • yummy ! wish they had <unk> at lunch and a dessert selection but a better value and greater than beef suggestion company . <EOS> | • hiliariously = ! food is not the best meal . bibachi cover service and they only wished a frozen yogurt for hot girl . better luck at <unk> and on the latter experience . <EOS> |
| • yummy ! wish they had <unk> at lunch and a dessert selection but a tiramisu better value and freshness value food taste better than iHop . <EOS> | • blended ! wifey better food ! the service is not frozen hot . they redeemed a <unk> and only frozen someplace at horse's for frozen worms . <EOS> |
| • yummy ! wish they had <unk> at lunch and a dessert selection but a tiramisu better value and freshness value food taste better than iHop . <EOS> | • wish ! wish they had <unk> at a buffet and netherlandish better tasting food . a renovation treasure and great value but not better than calories tasting . <EOS> |
| • as in china phoenix - this is one of the better ones . get there early - they fill up fast ! <EOS> | • wish ! wish they had <unk> at 10am and a dessert selection but better food a better and better tasting selection . great value ! <EOS> |
| • as far in san jose - this is one of the better ones . fast get up early ! there they fill up fast for u ! <EOS> | • wish ! wish they had lunch at <unk> and a dessert fountain but better than a selection and great tasting food servings better tasting . <EOS> |

**Table 16.** Interpolation results between latent codes of input sentences (with gray) from Yelp15.
| sMIM (1024) | sMIM (1024) |
|---|---|
| **(Business & Finance)** <EOS> are u shy or outgoing ? both , actually | are u shy or k ? both , actually <EOS> |
| ● are u or wishing video ? both , actually <EOS> | ● are u minded or rem ? actually , both <EOS> |
| ● are u stressed caffeine ? both , actually make a smile <EOS> | ● are u transparent or shy ? it’d actually , add-on <EOS> |
| ● witch are u or how lucky ? both <EOS> | ● are u uncontouchable cubed or program ?? both , actually like <EOS> |
| ● are u kidding or spraying ? both <EOS> | ● wha do u are roselle or marketed ? you start , by both my inbox <EOS> |
| ● how does wile or are you ? to both use , instead like it . <EOS> | ● how do u simplify phases towards you ? are proving , like no smiles . <EOS> |
| ● how do u choose to start or ? like i cant think , are actually better by my work . <EOS> | ● how do you start to rate ? i like kazzas when my was cheated . <EOS> |
| ● how do you start to alienate yourself ? i are like or drone , my actually feels . <EOS> | ● how do you start to start like i was taught by my parents . <EOS> |
| ● how do you start to yourself or like ? i like my math side . <EOS> | ● how do you start to like yourself ? i was taught by my parents . <EOS> |
| ● how do you start to like yourself ? i think my parents is by focusing . <EOS> | ● how do you start to cite yourself ? i like by my consequences in 1981 . <EOS> |
| ● how do you start to yourself like ? i was taught by my parents . <EOS> | ● how do you start to to beethoven ? like israel was my grandmother by fielders . <EOS> |

| (Health) | <EOS> how do you start to like yourself ? i was taught by my parents . |
|---|---|
| ● how do you start to yourself by allowing ? i like my parents yr . <EOS> | ● how do you start to like yourself ? i was taught by new england . <EOS> |
| ● how do you start to yourself like i ? my parents was by mario practitioner . <EOS> | ● how do you start to like yourself ? i was taught by my parents . <EOS> |
| ● how do you start to cite yourself ? i like by my consequences in 1981 . <EOS> | ● how do i start you to beethoven ? like israel was my grandmother by fielders . <EOS> |
| ● how do you start to yourself or like ? you can find yourself in my states , by today . <EOS> | ● how do you start to find ? i like aggland in my testicles was listening . <EOS> |
| ● how do you start yourself drunk ? i can find in something like to my country , what by jane . <EOS> | ● how can i do compuservre attain ? start to comment in spain you like , was my real pics . <EOS> |
| ● how can i start those need in america ? do you like to rephrase an invention , what i’m spinning ? <EOS> | ● how can i find blueprints do you ? i’m in spain like queens to chelsea , arrange . <EOS> |
| ● how can i find someone in spain ? i’m guessing today by pascal , what do you want to ? <EOS> | ● how can i find uneasy profiles in spain ? i’m sure what you do , like today’s ! <EOS> |
| ● how can i find an attorney in spain ? i’m studying chicken’s what , do you want to ? <EOS> | ● how can i find someone in spain ? i’m in spain today , what do you want ? <EOS> |
| ● how can i find someone in spain ? in spain i’m studying , what do you want ? <EOS> | ● how can i find someone in spain ? i’m in tanks today , what do you want to ? <EOS> |
| ● how can i find someone in spain ? i’m in italy today , what do you want ? <EOS> | ● how can i find someone in spain ? i’m guessing in spain today , what do you want ? <EOS> |

**Table 17.** Interpolation results between latent codes of input sentences (with gray) from *Yahoo Answers.*
D.3. Sampling

Table 19. Samples from best performing model for dataset Yelp15.

Table 20. Samples from best performing model for dataset Yahoo Answers.

Here we show samples from the best performing models learned from a single dataset for PTB (Fig. 18), Yelp15 (Fig. 19), and Yahoo Answers (Fig. 20). We sample from a zero-mean Gaussian distribution over the latent space, with an isotropic covariance with a standard deviation of 0.1 (since we cannot directly sample from the implicit marginal over the latent). Interestingly, this simple heuristic provides good samples. We attribution this to the anchor, which defines scale and position for the implicit marginal over the latent to roughly match.

D.4. Question Answering

Here we provide more examples of answers generated from a model trained on Yahoo Answers (i.e., sMIM (1024) in Fig. 21). In particular, the model was trained from data in which 20% of the encoder input tokens were replaced with the <unk> token. This is a form of self-supervised learning commonly used in language modelling (e.g., Bowman et al. (2015)). This encourages the model to replace <unk> with other tokens. We have found this procedure to significantly improve the quality of the generated answers. We provide three generated answers for each question (Q), taken from Yahoo Answers. Short/medium/long answers (A) are generated by concatenating 5/10/15 <unk> tokens. The number of <unk> encodes the length of the expected answer. We note that, in many cases, only one answer will be a good match to the question, suggesting the model has preferences towards answers with a question specific length.
Table 21. Question and sampled answers from model sMIM (1024) (i.e., trained on Yahoo Answers dataset). We provide short/medium/long sampled answers (A) for each question (Q).