Variational Autoencoders for Text Generation

Hareesh Bahuleyan

Borealis AI

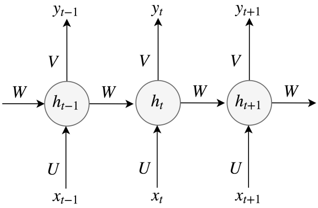
September 24, 2018

Overview

Plan

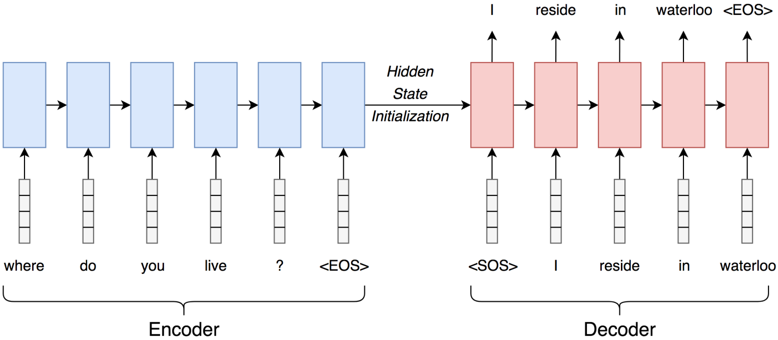
Recurrent Neural Networks

- Text data - expressed as a sequence
- RNNs
  - Feed inputs in a sequential manner
  - The hidden state contains info until t
  - $h_t = f(Ux_t + Wh_{t-1})$; $y_t = Vh_t$
  - Weight sharing

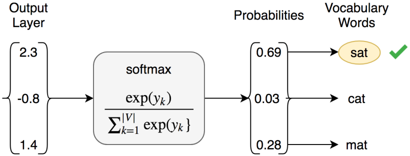


- Vanilla RNNs in practice
  - unable to remember the dependencies between inputs which are far apart in the sequence
- **Solution**: LSTM-RNNs [Hochreiter and Schmidhuber, 1997]
  - Better at capturing long term dependencies
  - An entire module (known as a *cell*) with a set of gates to replace f
  - Compute a hidden state h_t and a cell state c_t at each timestep

Sequence-to-Sequence Models



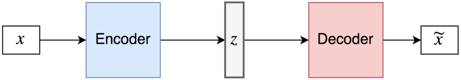
- Encoder and Decoder are RNNs with LSTM units
- Hidden state initialization
- Teacher Forcing
- Output Softmax layer

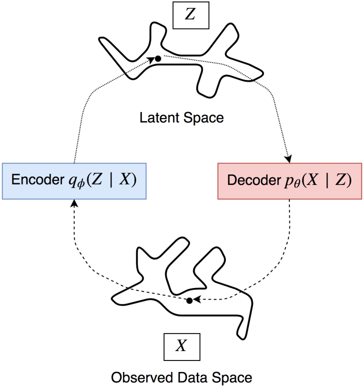


Plan

Autoencoding (Deterministic)

- Obtain a compressed representation of the data x from which it is possible to re-construct it
- Encoder $\boldsymbol{q_\phi}(\boldsymbol{z}|\boldsymbol{x})$ and Decoder $\boldsymbol{p_\theta}(\boldsymbol{x}|\boldsymbol{z})$ are jointly trained to maximize the conditional log-likelihood
- The latent representation $\boldsymbol{z}$ has an arbitrary distribution



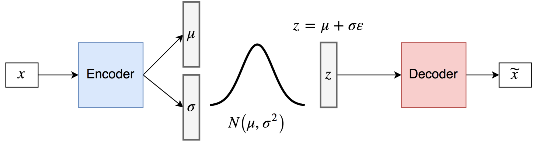


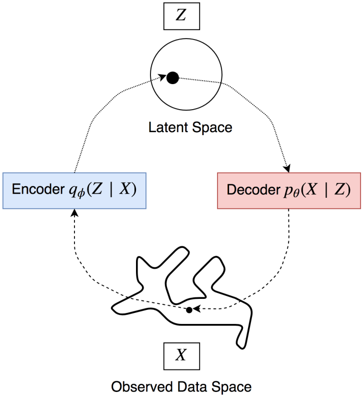
Minimize Reconstruction Loss

$$J = -\sum_{n=1}^{N} \sum_{t=1}^{|x^{(n)}|} \log p(x_t^{(n)}|z^{(n)}, x_{<t}^{(n)})$$

Variational Autoencoder [Kingma and Welling, 2013]

- Enforce a distribution on the latent space
- Minimize the Kullback-Leibler (KL) divergence between the learnt posterior and a pre-specified prior: $\mathrm{KL}(\mathcal{N}(\mu,\sigma)||\mathcal{N}(0,I))$
- Balance between reconstruction and KL penalty term
    - High λ - Ignores reconstruction
    - Low λ - Deterministic behaviour





Minimize Reconstruction Loss + KL Divergence

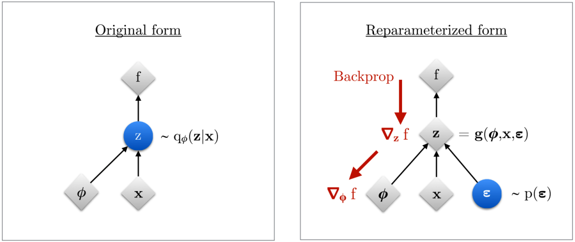
$$J = \sum_{n=1}^{N} \left[- \mathop{\mathbb{E}}_{z^{(n)}\sim q} \sum_{t=1}^{|x^{(n)}|} \log p(x_t^{(n)}|z^{(n)}, x_{<t}^{(n)}) + \lambda \cdot \mathrm{KL}(q(z^{(n)}|x^{(n)}) \| p(z)) \right]$$

Reparameterization Trick

KL Divergence between posterior and standard normal prior

$$\mathrm{KL}(\mathcal{N}(\mu,\sigma)||\mathcal{N}(0,I)) = \frac{1}{2}(1+\log((\boldsymbol{\sigma}^{(n)})^2) - (\boldsymbol{\mu}^{(n)})^2 - (\boldsymbol{\sigma}^{(n)})^2)$$

- Model training via SGD and error backpropagation
- Cannot sample directly from the approximate posterior distribution $\mathcal{N}(\mu,\sigma)$
- Stochastic Node - disconnect in the graph
- **Solution**: Sample from fixed distribution $\mathcal{N}(0,I)$ and reparameterize
- $\boldsymbol{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \otimes \epsilon$ where $\epsilon \sim \mathcal{N}(0,I)$



Training Heuristics

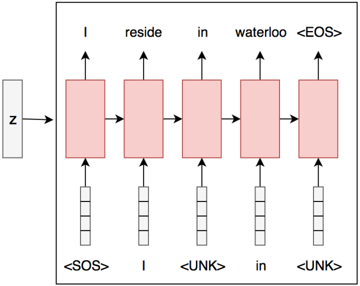
- Training VAEs for text generation is notoriously difficult
- Adopt two training strategies [Bowman et al., 2015]

KL Weight Annealing

- Gradually increase λ from zero to a threshold value
- Deterministic autoencoder $\rightarrow$ Variational autoencoder
- Experiment with different annealing schedules

Word Dropout

- Replace decoder inputs with $<$UNK$>$ with probability p
- Weakens the decoder and encourages the model to encode more information into z

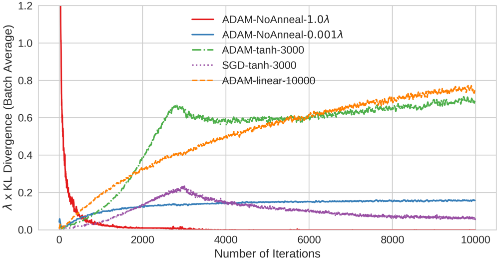


Decoder

VAE Variants

- Trained on 80k sentences of the SNLI dataset
- Evaluating reconstruction performance with BLEU scores
- BLEU-$j = \min\left(1, \frac{\text{generated-length}}{\text{reference-length}}\right) * (\text{precision}_j)$

Model	BLEU-4
Deterministic AE	73.73
ADAM-NoAnneal-1.0	2.05
ADAM-NoAnneal-0.001	72.05
ADAM-tanh-3000	36.50
SGD-tanh-3000	2.70
ADAM-linear-10000	35.29

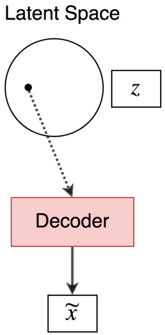


- Non-linear annealing $\lambda_i = \frac{\tanh\left(\frac{i-4500}{1000}\right)+1}{2}$
- Linear annealing $\lambda_i = \frac{i}{200000}$

Random Sampling

- VAEs exhibit interesting properties due to their learnt latent space
- Continuous latent space $\implies$ meaningful sentences
- Discard encoder; Sample from prior $\mathcal{N}(0, I)$ and generate
- New and interesting sentences unseen in the training data

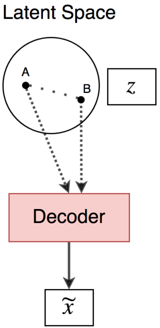
Deterministic AE	**ADAM-NoAnneal-1.0**
a men wears an umbrella waits to *a couple cows a monument* *there is sleeping and two rug .* *a man in a pick photos* *a boy are people at a lake escape .*	*a man is sitting on a bench .* *a man is sitting on a bench .* *a man is sitting on a bench .* *a man is sitting on a bench .* *a man is sitting on a bench .*
ADAM-NoAnneal-0.001	**ADAM-tanh-3000**
i woman who is on watch a factory *they are excited formation to ride a castle of a* *their janitor is leaving the dirt wearing his suits .* *two children in it exits a* *six people sitting are sorting at single radio in .*	*the dog is sleeping in the grass .* *the girls are being detained .* *the group of people are going to begin .* *a girl with blond-hair on a bike with a stick* *a woman and a man are walking on a street*



Linear Interpolation

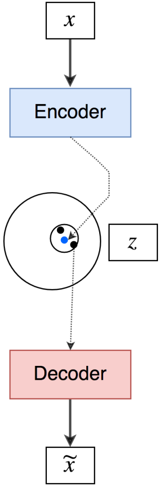
- To test the continuity of the latent space
- $\mathbf{z}_{\alpha_i} = \alpha_i \cdot \mathbf{z}_A + (1 - \alpha_i) \cdot \mathbf{z}_B$ where $\alpha_i \in \left[0, \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}, 1\right]$
- VAE - Smooth transition maintaining syntax and semantics
- DAE - Transition is irregular and non-continuous

Deterministic AE	Variational AE
Sentence A: there is a couple eating cake .	
there is a couple eating cake .	*there is a couple eating cake .*
there is a couple eating cake .	*there is a couple eating .*
there is a couple eating cake .	*there is a couple eating dinner .*
there is a group of people eating a party .	*there is a couple of people eating dinner .*
a group of men are watching a party .	*a group of people are having a conversation .*
a group of men are watching a dance party .	*a group of men are having a discussion .*
a group of men are watching a dance party .	*a group of men are watching a movie .*
a group of men are watching a dance party .	*a group of men are watching a movie theater .*
Sentence B: a group of men are watching a dance party .	



Sampling from Neighborhood

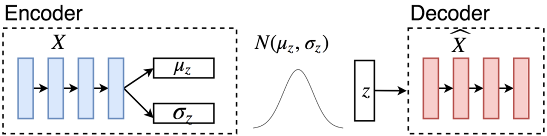
- For a given input $\boldsymbol{x}$, sample the latent vector as $\boldsymbol{z} = \boldsymbol{\mu} + 3\boldsymbol{\sigma} \otimes \epsilon$
- VAE - generates diverse sentences, however topically similar to the input.
- DAE - latent space has empty regions



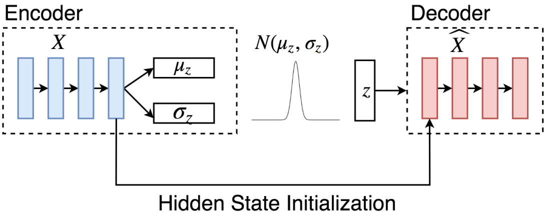
Deterministic AE	**Variational AE**
Input Sentence: a dog with its mouth open is running .	
a dog with its mouth is open running .	*a dog with long hair is eating .*
a dog with its mouth is open running .	*a guy and the dogs are holding hands*
a dog with its mouth is open running .	*a dog with a toy at a rodeo .*
Input Sentence: there are people sitting on the side of the road	
there are people sitting on the side of the road	*the boy is walking down the street .*
there are people sitting on the side of the road	*there are people standing on the street outside*
there are people sitting on the side of the road	*the police are on the street corner .*

VAE Bypassing Phenomenon

- Design considerations
- z is sampled and fed to the decoder
- Encode useful information in the latent space



- With **bypass connection**, the decoder has direct deterministic access to the source info
- Latent space ignored, KL divergence doesn't act as a regularizer



Diversity Evaluation Metrics

For a given input $\boldsymbol{x}$, generate multiple outputs $\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_k$

Entropy

- Compute unigram probability $p(w)$ of each word in the generated set
- $H = -\sum_w p(w) \log p(w)$
- More entropy $\implies$ more randomness $\implies$ more diversity

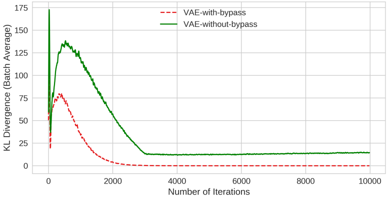
Distinct Scores

- $\text{Distinct-1} = \frac{\text{Count of distinct unigrams}}{\text{Total unigram count}}$
- $\text{Distinct-2} = \frac{\text{Count of distinct bigrams}}{\text{Total bigram count}}$

Effect on Latent Space

- VAE without hidden state initialization generates diverse outputs
- Bypass connection degrades the model to a deterministic AE

	VAE with Bypass	VAE without Bypass
Entropy	2.004	2.686
Distinct-1	0.099	0.302
Distinct-2	0.118	0.502



VAE with Bypass	VAE without Bypass
Input Sentence: the men are playing musical instruments	
the men are playing musical instruments *the man is playing musical instruments* *the men are playing musical instruments*	*the men are playing video games* *the men are playing musical instruments* *the musicians are playing musical instruments*
Input Sentence: a child holds a shovel on the beach .	
a child holds a shovel on the beach . *a child holds a shovel on the beach .* *a child holds a shovel on the beach .*	*a child playing with the ball on the beach .* *a child holding a toy on the water .* *a child holding a toy on the beach .*

Plan

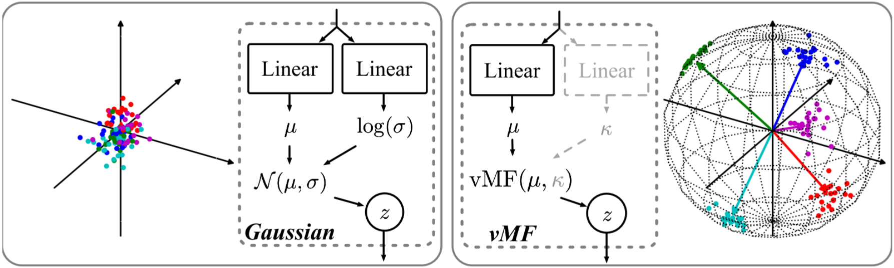
Problem Addressed

- VAE with multivariate Gaussian prior and posterior has issues associated with KL term collapsing to zero (especially for text generation)
- Instead use a **von Mises-Fisher** distribution to circumvent this issue Davidson et al. [2018], Xu and Durrett [2018]

von Mises-Fisher Distribution

Places a distribution over a **unit** hypersphere

- mean μ, which acts as a location parameter (denotes where in the hyperspace is it located), magnitude needs to be 1, since its a unit sphere
- concentration parameter κ - the concentration of data probability happens in the direction of μ

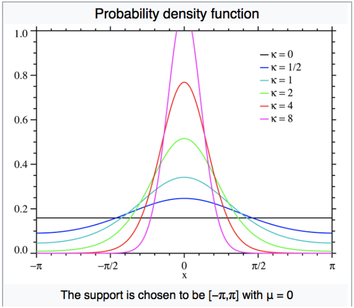


von Mises-Fisher Distribution

$$f_d(x; \mu, \kappa) = C_d(\kappa) \exp(\kappa \mu^T x)$$

$$C_d(\kappa) = \frac{\kappa^{d/2-1}}{(2\pi)^{d/2} I_{d/2-1}(\kappa)}$$

- $\kappa = 0$ corresponds to a uniform distribution of datapoints on the sphere
- Larger values of κ corresponds to more 'normal-like' distribution of the data points.



Spherical VAE Model Details

- Prior: uniform distribution vMF(.,$\kappa = 0$)
- Posterior: vMF(μ, κ = fixed) - Only μ is learnt from the encoder output

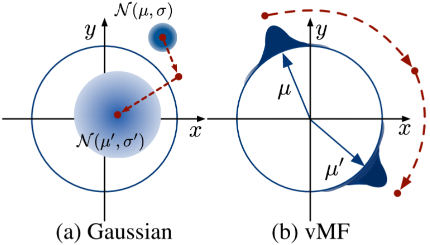


Figure 2: Visualization of optimization of how q varies over time for a single example during learning. In the Gaussian case, the KL term tends to pull the model towards the prior (moving from μ, σ to μ', σ'), whereas in the vMF case there is no such pressure towards a single distribution.

vMF KL Divergence

KL divergence With vMF$(\cdot, 0)$ as our prior, the KL divergence is:[4]

$$\mathrm{KL}(\mathrm{vMF}(\mu,\kappa)||\mathrm{vMF}(\cdot,0)) = \kappa \frac{I_{d/2}(\kappa)}{I_{d/2-1}(\kappa)} + \left(\frac{d}{2}-1\right)\log\kappa - \frac{d}{2}\log(2\pi) - \log I_{d/2-1}(\kappa) + \frac{d}{2}\log\pi + \log 2 - \log\Gamma\left(\frac{d}{2}\right)$$

- Critically, this only depends on κ, not on μ.
- κ will be treated as a fixed hyperparameter
- This is strange since it is now like a DAE, since we have the objective function with reconstruction loss + a constant (KL term)
- The KL term only depends on the dimensionality of the latent space!

Model Details

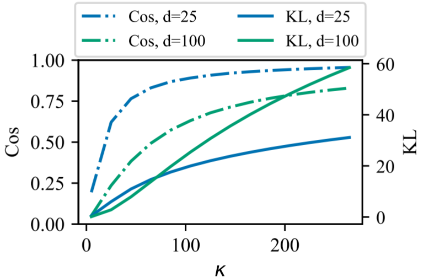


Figure 3: Visualization of the interaction between κ, KL, and dimensionality in vMF. Cos represents the cosine similarity between μ and samples from $\text{vMF}_d(\mu, \kappa)$ which reflects how disperse the distribution is. KL is defined as KL with a uniform vMF prior, $\text{KL}(\text{vMF}_d(\mu, \kappa)||\text{vMF}(\cdot, 0))$. Higher κ values yield higher cosine similarities, but also higher KL costs.

- Higher dimension $\implies$ higher *constant* KL loss
- Higher κ values yield higher cosine similarities - points are concentrated just around the mean (meaning **less disperse** - is that even desirable?)

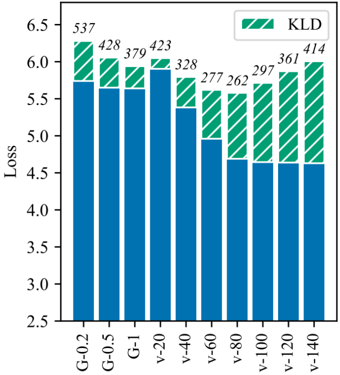
Language Modelling Results

Evaluate using PPL and NLL, Compare to Bowman et al. [2015]

Model	PTB				Yelp			
	Standard		Inputless		Standard		Inputless	
	NLL	PPL	NLL	PPL	NLL	PPL	NLL	PPL
RNNLM (2016)	100 (–)	116	135 (–)	>600	–	–	–	–
G-VAE (2016)	101 (2)	119	125 (15)	380	–	–	–	–
RNNLM (Ours)	100 (–)	114	134 (–)	596	199 (–)	55	300 (–)	432
G-VAE (Ours)	99 (4.4)	109	125 (6.3)	379	199 (0.5)	55	274 (13.4)	256
vMF-VAE (Ours)	**96 (5.7)**	**98**	**117 (18.6)**	**262**	**198 (6.4)**	**54**	**242 (48.5)**	**134**

KL Divergence Comparison

Able to obtain non-zero values of KL divergence (and hence a good latent space), without any dirty engineering tricks



What does the Latent Space Learn ?

- **Experiment**
  - To understand what the vMF-VAE encodes
  - Compute BoW vector = average of word embeddings
  - At each time step, they pass on BoW vector along with latebt vector $\boldsymbol{z}$
- When additional information provided in the form of BoWs, the Gaussian-VAE latent space collapses (since model can choose to ignore it), while their model still performs good
- Another justification to show that their model's latent space learns more than just a simple BoWs
  - Word ordering, better semantics, etc.

Model	NVRNN		NVRNN-BoW
Setting	$\mu \rightarrow$ BoW	BoW $\rightarrow \mu$	$\mu \rightarrow$ BoW
G-VAE	0.74	0.74	0.32
v-VAE	0.77	0.57	0.23

Plan

1. Background
2. Variational Autoencoder
3. Spherical VAEs
4. Conclusions

Summary and Discussions

- VAEs are generative models from which it is possible to synthesize new data
- Text generation in VAEs are notoriously difficult due to issues associated with KL loss vanishing to zero
- Spherical VAEs address this issue by using vMF distribution instead of Gaussian
- Some critical views:
  - In vMF-VAE, the KL loss terms just acts as an additive constant to the objective function
  - Fixing κ seems to work better than allowing it to be learnt, their claim is when κ is learnable, the KL term will encourage κ to be as low as possible - isn't this because their prior has a low κ value, i.e., $\kappa = 0$? - No experimental result shown for this!
  - No qualitative analysis of latent space - linear interpolation, random sampling as in Bowman et al. [2015]

References I

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*, 2015.

Tim R Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M Tomczak. Hyperspherical variational auto-encoders. *arXiv preprint arXiv:1804.00891*, 2018.

Jiacheng Xu and Greg Durrett. Spherical latent spaces for stable variational autoencoders. *arXiv preprint arXiv:1808.10805*, 2018.