Von Mises-Fisher Loss For Training Sequence To Sequence Models With Continuous Outputs

ICLR 2019

Kunmar et al.

Presenter: Jungsoo Park

Data Mining & Information Systems Lab.
Department of Computer Science and Engineering,
College of Informatics, Korea University

1 Introduction

Time Matters

As for BERT

Pre-Training took 4 days using

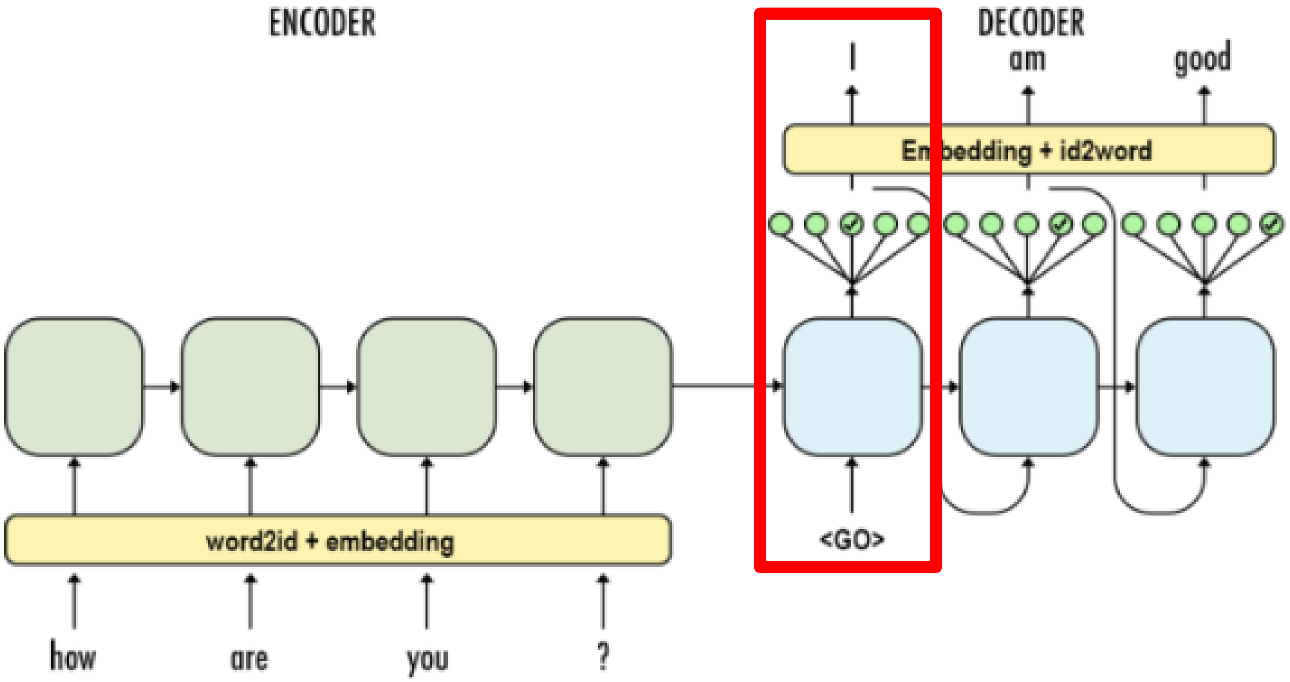
- BERT-Base: 4 Cloud TPUs (16 TPU chips total)
- BERT-Large: 16 Cloud TPUs (64 TPU chips total)

If using 8 TESLA P100,

it would take one year in training

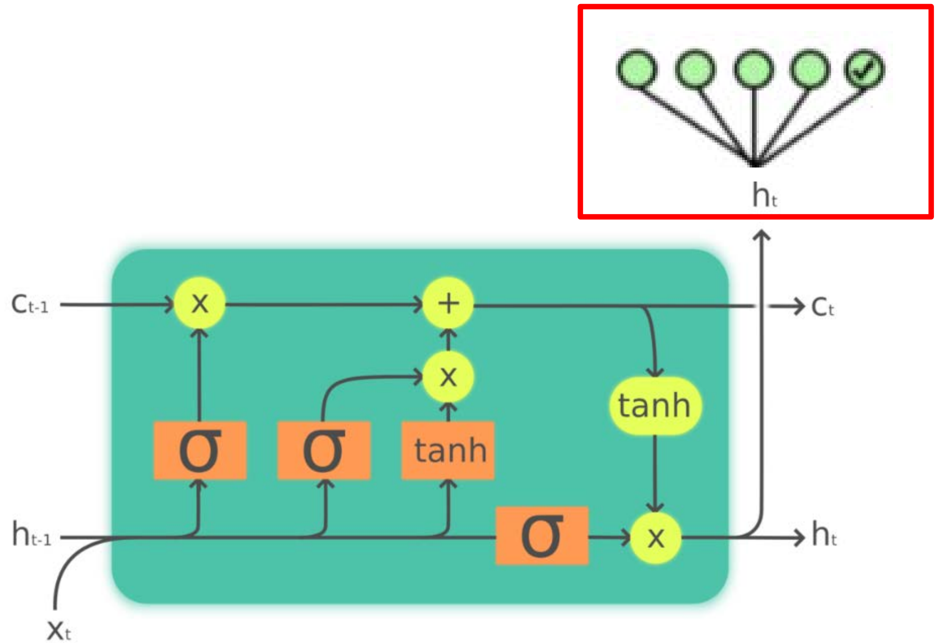
1 Introduction

Seq2Seq



1 Introduction

Decoding



Bottleneck in training

$$s_w = W_{hw}\mathbf{h}_t + b_w$$
$$W \in \mathbb{R}^{V\text{x}H}$$
$$b \in \mathbb{R}^v$$

Bottleneck in training

$$\boxed{\mathbf{p}_t(w) = \frac{e^{s_w}}{\sum_{v\in\mathcal{V}} e^{s_v}}}$$

$$\mathrm{NLL}(\mathbf{p_t}, \mathbf{o}(w)) = -\log(\mathbf{p_t}(w))$$

1 Introduction

Byte Pair Encoding

Initial Vocab

```
low : 5, lower : 2, newest : 6, widest : 3
```

Characterize

```
# dictionary
l o w : 5,  l o w e r : 2,  n e w e s t : 6,  w i d e s t : 3
```

```
# vocabulary
l, o, w, e, r, n, w, s, t, i, d
```

```
l o w : 5,
l o w e r : 2,
n e w es t : 6,
w i d es t : 3
```

⇨

```
l o w : 5,
l o w e r : 2,
n e w est : 6,
w i d est : 3
```

⇨

```
lo w : 5,
lo w e r : 2,
n e w est : 6,
w i d est : 3
```

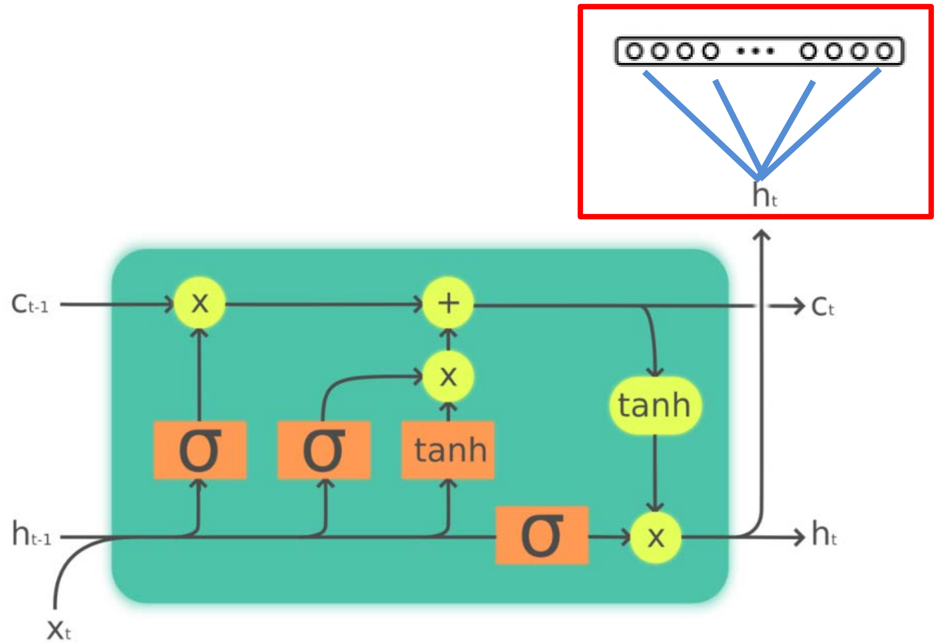
⇨

```
low : 5,
low e r : 2,
newest : 6,
widest : 3
```

```
# vocabulary update!
l, o, w, e, r, n, w, s, t, i, d, es, est, lo, low, ne, new, newest, wi, wid, widest
```

2 Method

Decoding



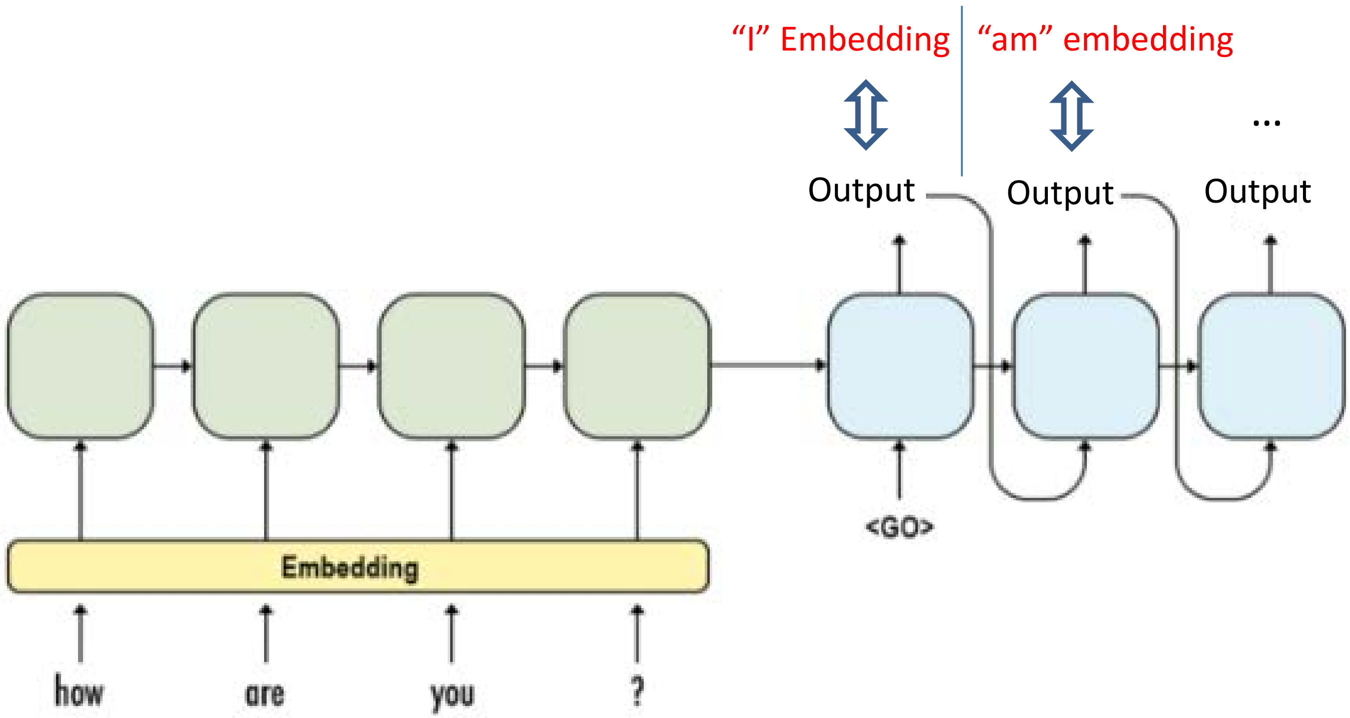
Predicting
Embedding Vector

No need for soft-max operation, but
predict the word embedding vector itself

2 Method

Overview

Loss occurs from the target word(label's) embedding vector and our generated one's



2 Method

Pros

1. Big Batch Size

2. Unique Approach to Handle Bottleneck in Softmax

Cons

1. Beam Search Not Available (KNN)

2. Inference Not Efficient

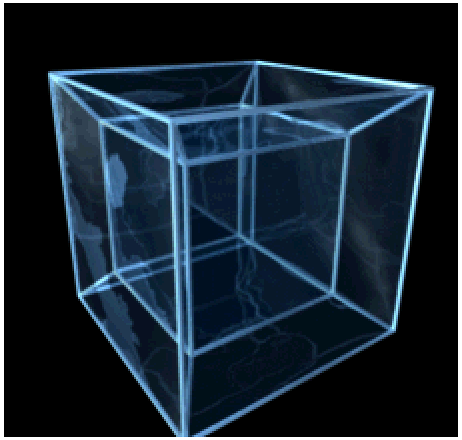
3. Lack of Experiments based on Contextualized Representation

Loss Function

But what loss function should we use?

L2 Loss?

Basic assumption behind using L2 loss is that
the output space follows Gaussian distribution
and is a Distance-based metric

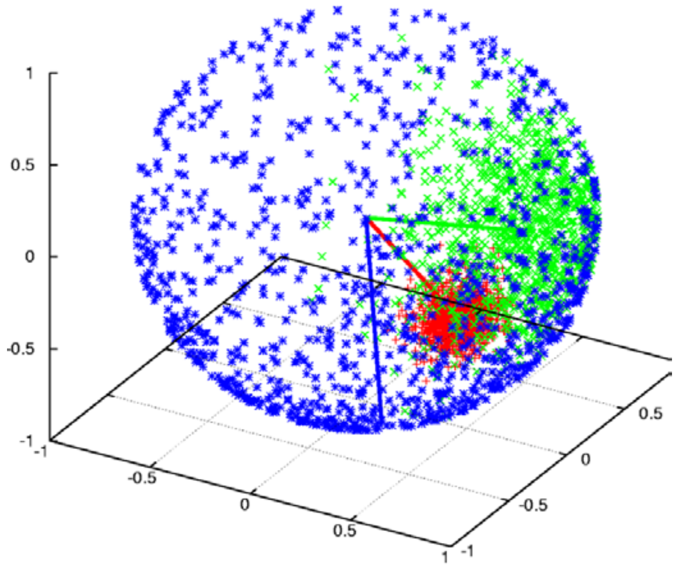


Hypercube

"Most of the volume of a high-dimensional orange is in the skin, not the pulp."

https://homes.cs.washington.edu/~pedrod/papers/cacm12.pdf https://www.youtube.com/watch?v=zwAD6dRSVyI&t=2s

Therefore, it's not the best way to use distance metric,
but instead use direction based metric for high dimension data



Von Mises-Fisher

Directional Equivalent of Gaussian
Defined on the Support of (p-1)
Dimensional Sphere

$$p(\mathbf{e}(w); \boldsymbol{\mu}, \kappa) = C_m(\kappa) e^{\kappa \boldsymbol{\mu}^T \mathbf{e}(w)}$$

$$C_m(\kappa) = \frac{\kappa^{m/2-1}}{(2\pi)^{m/2} I_{m/2-1}(\kappa)},$$

$$p(\mathbf{e}(w); \hat{\mathbf{e}}) = \mathrm{vMF}(\mathbf{e}(w); \hat{\mathbf{e}}) = C_m(\|\hat{\mathbf{e}}\|) e^{\hat{\mathbf{e}}^T \mathbf{e}(w)}$$

$$\boxed{\mathrm{NLLvMF}(\hat{\mathbf{e}}; \mathbf{e}(w)) = -\log\left(C_m(\|\hat{\mathbf{e}}\|)\right) - \hat{\mathbf{e}}^T \mathbf{e}(w)}$$

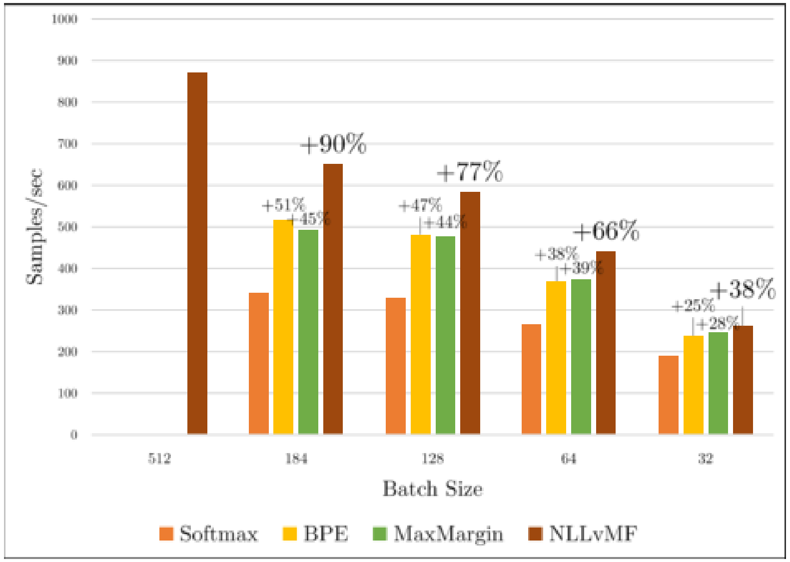
https://sgfin.github.io/2017/03/16/Deriving-probability-distributions-using-the-Principle-of-Maximum-Entropy/

4 Experiment

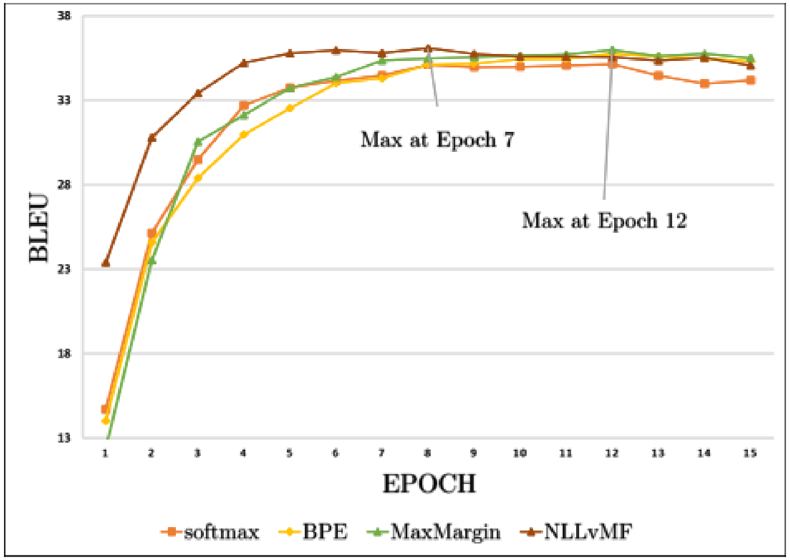
Embedding Model	Tied Emb	Source Type/ Target Type	Loss	BLEU		
				fr–en	de–en	en–fr
-	no	word→word	CE	31.0	24.7	29.3
-	no	word→BPE	CE	29.1	24.1	29.8
-	no	BPE→BPE	CE	**31.4**	**25.8**	**31.0**
word2vec	no	word→emb	L2	27.2	19.4	26.4
word2vec	no	word→emb	Cosine	29.1	21.9	26.6
word2vec	no	word→emb	MaxMargin	29.6	21.4	26.7
fasttext	no	word→emb	MaxMargin	31.0	25.0	29.0
fasttext	yes	word→emb	MaxMargin	**32.1**	**25.0**	**31.0**
word2vec	no	word→emb	$\text{NLLvMF}_{\text{reg1}}$	29.5	22.7	26.6
word2vec	no	word→emb	$\text{NLLvMF}_{\text{reg1+reg2}}$	29.7	21.6	26.7
word2vec	yes	word→emb	$\text{NLLvMF}_{\text{reg1+reg2}}$	29.7	22.2	27.5
fasttext	no	word→emb	$\text{NLLvMF}_{\text{reg1+reg2}}$	30.4	23.4	27.6
fasttext	yes	word→emb	$\text{NLLvMF}_{\text{reg1+reg2}}$	**32.1**	**25.1**	**31.7**

- x2.5 times faster than training word -> word (baseline) but shows comparable result
- Proposed Loss results in better result than the empirical loss functions(L2, Cosine Loss, Max Margin Loss)

4 Experiment



Samples processed per second
(Proposed method outperforms)



Convergence
(NLLvMF Loss outperforms in stability)