Parameter-free Sentence Embedding via Orthogonal Basis

Ziyi Yang[1]*, Chenguang Zhu[2], and Weizhu Chen[3]

[1]Department of Mechanical Engineering, Stanford university
[2]Microsoft Speech and Dialogue Research Group
[3]Microsoft Dynamics 365 AI
ziyi.yang@stanford.edu, {*chezhu, wzchen*}*@microsoft.com*

(EMNLP2019)

Presenter: Naoya Inoue

Tohoku University / RIKEN AIP

※図表は論文より引用

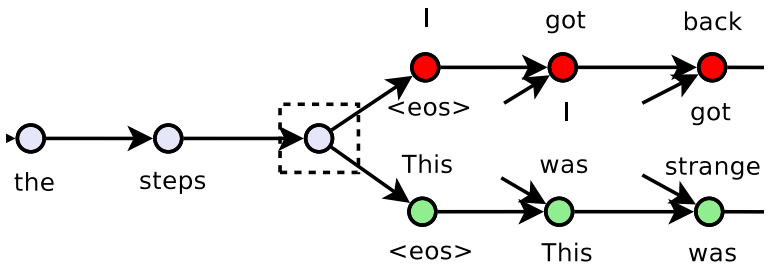
Sentence embedding

- Vector representation of sentence
  - s = "Don't think, feel."
    → $\boldsymbol{c_s}$ =[3.21, 1.23, 0.85, …, 3.41]
- Desirable properties (Bowman+ Tutorial @ *SEM2019)
  - Word contents and word order.
  - (Rough) grammatical structure.
  - Cues to connotation and social meaning.
  - Unambiguous propositional information (of the kind expressed in a semantic parse).

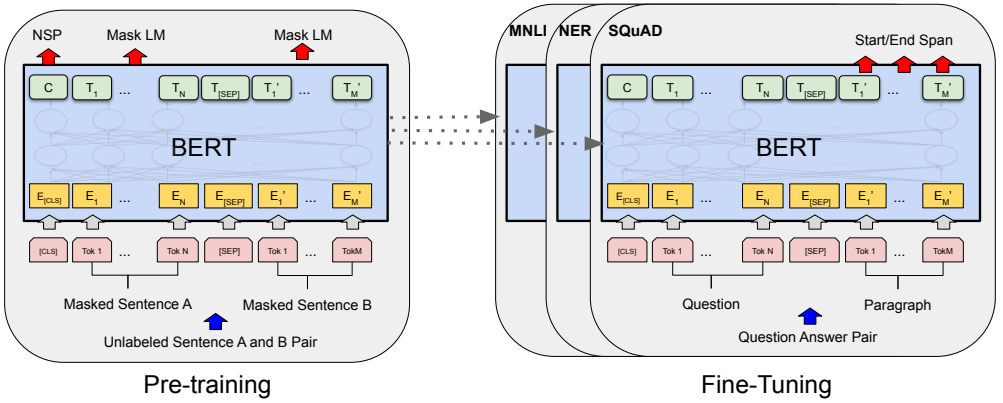
    $$\forall x[\text{patient}'(x) \rightarrow \exists y[\text{doctor}'(y) \wedge \text{treat}'(y, x)]]$$
- Applications: everywhere

Previous approaches: parametrized

- Encoder is parametrized and requires training
  - SkipThought (Kiros+2017), Sent2Vec (Pagliardini+2018), QuickThought (Logeswaran+2018), *à la carte* (Khodak+2018), InferSent (Conneau+ 2017), Universal Sentence Encoder (Cer+2018), ELMo (Peters+2017), BERT (Devlin+2019) etc.



SkipThought (Kiros+2017)



BERT (Devlin+2019)

Previous approaches: parameter-free

- Encoder has no parameters and requires NO further training upon pretrained word embeddings
  - SIF (Arora+2017), uSIF (Ethayarajh+2018), p-mean (Ruckle+2018)

1: **for all** sentence s in $\mathcal{S}$ **do**
2: $v_s \leftarrow \frac{1}{|s|} \sum_{w \in s} \frac{a}{a+p(w)} v_w$
3: **end for**
4: Form a matrix X whose colur
5: **for all** sentence s in $\mathcal{S}$ **do**
6: $v_s \leftarrow v_s - uu^\top v_s$
7: **end for**

SIF (Arora et al. 2017)

$$\left(\frac{x_1^p + \cdots + x_n^p}{n} \right)^{1/p} ; \quad p \in \mathbb{R} \cup \{\pm\infty\}$$

$$\mathbf{s}^{(i)} = H_{p_1}(\mathbf{W}^{(i)}) \oplus \cdots \oplus H_{p_K}(\mathbf{W}^{(i)})$$

$$\bigoplus_i \mathbf{s}^{(i)}$$

p-mean (Ruckle+2018)

This work: **G**eometric **EM**bedding (GEM)

- **Key idea**
  - Parameter-free
    - ✓ Easy adaptation to novel domain
    - ✓ Fast inference speed
  - Sentence embedding = Weighted sum of word embeddings
    - Weight = Novelty (**新規性**) + Significance (**重要性**)
      + Corpus-wise uniqueness (**コーパス全体からみた独自性**; IDF)
- **Key contribution**
  - New way to quantify semantic meaning (above) of words in sentences via orthogonal basis
  - Outperforms all previous parameter-free encoders (except for uSIF)

Sentence embedding in GEM

- Defines embedding $\boldsymbol{c_s}$ of sentence $\mathbf{s} = (w_1, w_2, \ldots, w_n)$:

$$\boldsymbol{c}_s = \sum_i \alpha_i \boldsymbol{v}_{w_i} \qquad (10)$$

$$\alpha_i = \alpha_n + \alpha_s + \alpha_u$$

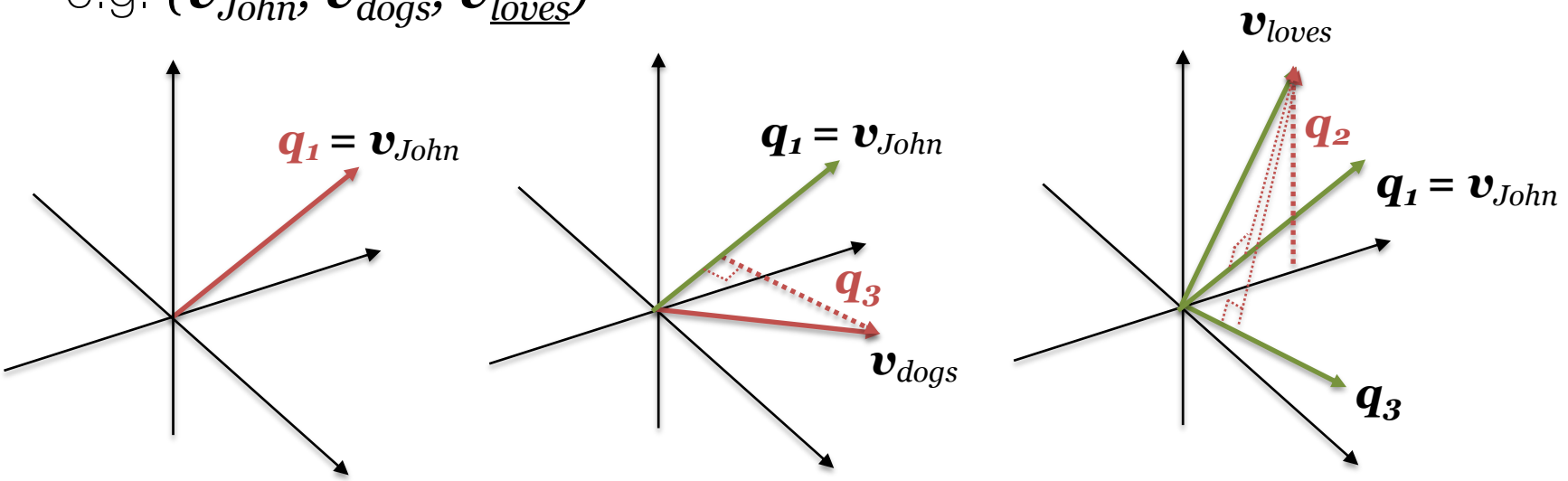
  - $\boldsymbol{v}_{w_i}$: Pretrained word embedding of $\boldsymbol{w_i}$
  - α_n: Novelty of $\boldsymbol{w_i}$ wrt contextual window ($\boldsymbol{m}$ words)
  - α_s: Significance of $\boldsymbol{w_i}$ wrt contextual window ($\boldsymbol{m}$ words)
  - α_u: Uniqueness of $\boldsymbol{w_i}$ in corpus

Contextual window matrix

- Consider a sequence of contextual word embeddings $\boldsymbol{w_i}$ concatenated with $\boldsymbol{v_{w_i}}$:
  - $(\boldsymbol{v_{w_{i-m}}}, \cdots, \boldsymbol{v_{w_{i-1}}}, \boldsymbol{v_{w_{i+1}}}, \cdots, \boldsymbol{v_{w_{i+m}}}, \boldsymbol{v_{w_i}})$
  - $2m + 1$ vectors
  - e.g. “John <u>loves</u> dogs.”, ($\boldsymbol{i=2}$) $\rightarrow (\boldsymbol{v_{John}}, \boldsymbol{v_{dogs}}, \boldsymbol{v_{\underline{loves}}})$

Constructing orthogonal basis

- Construct orthogonal basis $q_{i-m}, \ldots, q_{i+m}, q_i$ using Gram-Schmidt Process
  - e.g. $(\boldsymbol{v}_{John}, \boldsymbol{v}_{dogs}, \boldsymbol{v}_{\underline{loves}})$

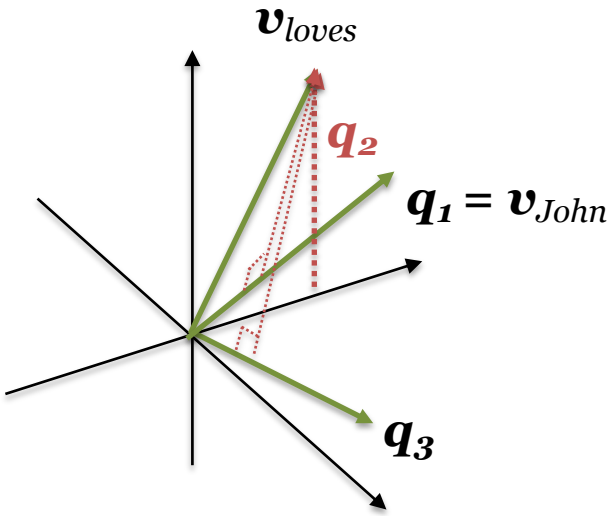


- Use q_i to see relation between hyperplane spanned by $q_{i-m}, \ldots, q_{i+m}$ (*context hyperplane*) and $\boldsymbol{v}_{w_i}$

Novelty

- Normalized distance from context hyperplane to word embedding ($\mathbf{sin\theta}$)

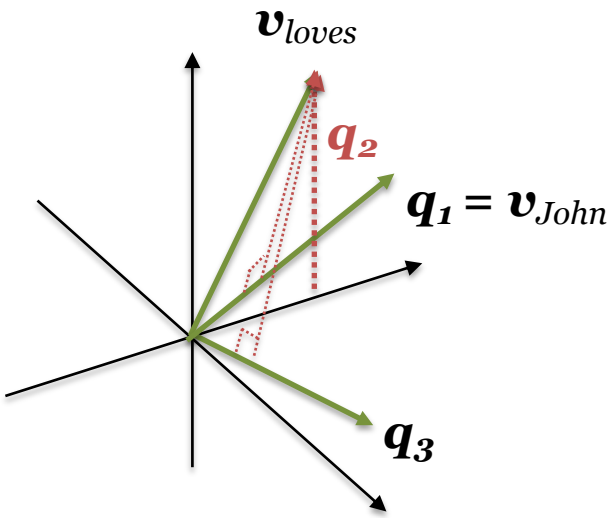
$$\alpha_n = \exp\left(\frac{\|\boldsymbol{q}_i\|_2}{\|\boldsymbol{v}_{w_i}\|_2}\right)$$



Significance

- Absolute distance from context hyper-plane to word embedding

$$\alpha_s = \frac{\|\boldsymbol{q_i}\|_2}{2m+1}$$



Corpus-wise uniqueness

- Similar idea to Inverse Document Frequency (IDF)
  - Common words (e.g. the, a, is) → uniqueness ↓
- Offline processing:
  - Calculate $\boldsymbol{K}$ corpus-wide principal vectors $\{\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_\mathrm{K}\}$
- Sentence-dependent processing:
  - Rank them according to correlation with sentence embedding $\boldsymbol{S} = [\boldsymbol{v}_{w_1}, \boldsymbol{v}_{w_2}, \ldots, \boldsymbol{v}_{w_n}]$
  - Take top $\boldsymbol{h}$ vectors $\boldsymbol{D} = [\boldsymbol{d}_{\mathrm{t1}}, \boldsymbol{d}_{\mathrm{t2}}, \ldots, \boldsymbol{d}_{\mathrm{th}}]$ (with singular values $\boldsymbol{\sigma}_d$)

$$\alpha_u = \exp\left(-\|\boldsymbol{\sigma}_d \odot (\boldsymbol{q}_i^T \boldsymbol{D})\|_2 / h\right)$$

おまけ: Sentence-dependent removal of principle components (SDR)

- Sentence-independent removal (Arora+2017)

1: **for all** sentence s in $\mathcal{S}$ **do**
2: $v_s \leftarrow \frac{1}{|s|} \sum_{w \in s} \frac{a}{a+p(w)} v_w$
3: **end for**
4: Form a matrix X whose colur
5: **for all** sentence s in $\mathcal{S}$ **do**
6: $v_s \leftarrow v_s - uu^\top v_s$
7: **end for**

u: first principal component

  - Problem: suboptimal, each sentence has different meanings

- This work:

$$\boldsymbol{c}_s \leftarrow \boldsymbol{c}_s - \boldsymbol{D}\boldsymbol{D}^T \boldsymbol{c}_s$$

Experiments

- Pretrained word embeddings
  - LexVec, fastText, PSL (Wieting+ 2015)
- Hyper-parameters
  - $m = 7$ (size of contextual window)
  - $K = 45$ (# corpus-wide principal vectors)
  - $h = 17$ (# sentence-dependent principal vectors)
- Testbed
  - Unsupervised textual similarity task
    - STS: semantic textual similarity (sentence pair → sim. Score)
    - CQA: given a community question, rank 10 questions based on sim.
  - Supervised task (sentiment analysis, textual entailment etc.)
    - Linear classifier is learned on top of GEM embedding

Results: STS, CQA

Non-parameterized models	dev	test
GEM + L.F.P (ours)	83.5	78.4
GEM + LexVec (ours)	81.9	76.5
SIF (Arora et al., 2017)	80.1	72.0
uSIF (Ethayarajh, 2018)	**84.2**	79.5
LexVec	58.78	50.43
L.F.P	62.4	52.0
word2vec skipgram	70.0	56.5
Glove	52.4	40.6
ELMo	64.6	55.9
Parameterized models		
PARANMT-50M (Wieting and Gimpel, 2017a)	-	**79.9**
Reddit + SNLI (Yang et al., 2018)	81.4	78.2
GRAN (Wieting and Gimpel, 2017b)	81.8	76.4
InferSent (Conneau et al., 2017)	80.1	75.8
Sent2Vec (Pagliardini et al., 2018)	78.7	75.5
Paragram-Phrase (Wieting et al., 2015a)	73.9	73.2

Table 1: Pearson's $r \times 100$ on STSB. Best results are in bold.

GEM + L.F.P (ours)	**49.11**
Reddit + SNLI tuned	47.44
KeLP-contrastive1	49.00
SimBow-contrastive2	47.87
SimBow-primary	47.22

Table 2: MAP on CQA subtask B.

- STS
  - Outperforms parameter-free models except for uSIF
    - uSIF depends on prior knowledge statistics
  - Close to SOTA parameterized model
- CQA
  - Outperforms or comparable to parametrized models trained on training dataset

Results: sentence embedding benchmark

Model	Dim	Training time (h)	MR	CR	SUBJ	MPQA	SST	TREC	MRPC	SICK-R	SICK-E
Non-parameterized models											
GEM + L.F.P	900	0	**79.8**	**82.5**	**93.8**	**89.9**	**84.7**	**91.4**	**75.4/82.9**	**86.5**	**86.2**
GEM + GloVe	300	0	78.8	81.1	93.1	89.4	83.6	88.6	73.4/82.3	86.3	85.3
SIF	300	0	77.3	78.6	90.5	87.0	82.2	78.0	-	86.0	84.6
uSIF	300	0	-	-	-	-	80.7	-	-	83.8	81.1
p-mean	3600	0	78.4	80.4	93.1	88.9	83.0	90.6	-	-	-
GloVe BOW	300	0	78.7	78.5	91.6	87.6	79.8	83.6	72.1/80.9	80.0	78.6
Paraemterized models											
InferSent	4096	24	81.1	86.3	92.4	90.2	84.6	88.2	76.2/83.1	88.4	86.3
Sent2Vec	700	6.5	75.8	80.3	91.1	85.9	-	86.4	72.5/80.8	-	-
SkipThought-LN	4800	336	79.4	83.1	93.7	89.3	82.9	88.4	-	85.8	79.5
FastSent	300	2	70.8	78.4	88.7	80.6	-	76.8	72.2/80.3	-	-
à la carte	4800	N/A	81.8	84.3	93.8	87.6	86.7	89.0	-	-	-
SDAE	2400	192	74.6	78.0	90.8	86.9	-	78.4	73.7/80.7	-	-
QT	4800	28	82.4	86.0	**94.8**	90.2	**87.6**	92.4	76.9/84.0	87.4	-
STN	4096	168	**82.5**	**87.7**	94.0	**90.9**	83.2	93.0	**78.6/84.4**	**88.8**	**87.8**
USE	512	N/A	81.36	86.08	93.66	87.14	86.24	**96.60**	-	-	-

Table 3: Results on supervised tasks. Sentence embeddings are fixed for downstream supervised tasks. Best results for each task are underlined, best results from models in the same category are in bold. SIF results are extracted from Arora et al. (2017) and Rücklé et al. (2018), and training time is collected from Logeswaran and Lee (2018).

- Outperforms all parameter free models, comparable to supervised models in some tasks

Results: Ablation study in STS

	Configurations	STSB dev	SUBJ
①	Mean of L.F.P	62.4	-
①	GEM weights	71.0	-
①	GEM weights + SIR	81.8	-
①	GEM weights + SDR	83.5	-
②	α_n + SDR	81.6	93.42
②	α_n, α_s + SDR	81.9	93.6
③	$\alpha_n, \alpha_s, \alpha_u$ + SDR	83.5	93.8

- ① Removal of corpus-wide principal components is important (consistent with Arora+2017)
  - Sentence-dependent removal makes it more effective
- ② Significance is not effective
- ③ Uniqueness is important

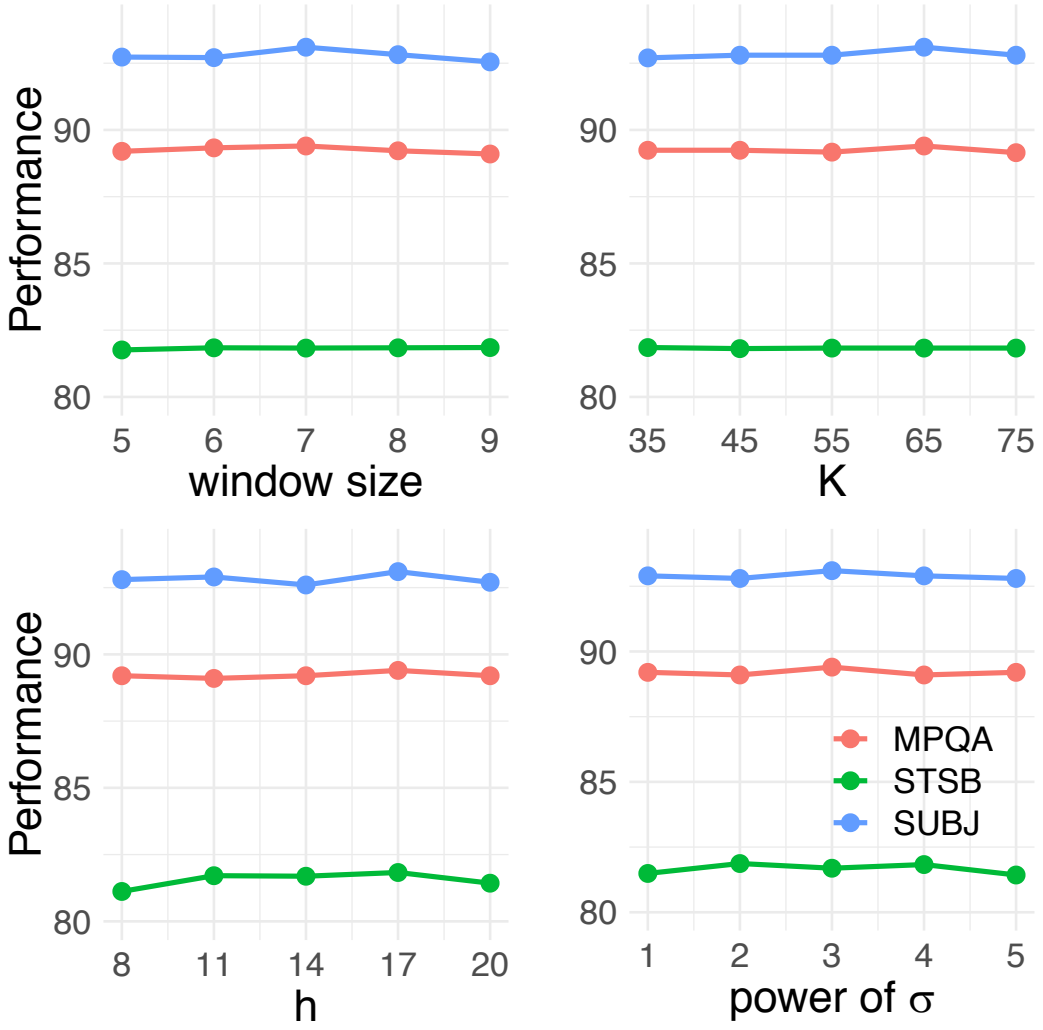
Results: Inference speed

	Average run time (s)	Variance
GEM (CPU)	**20.08**	0.23
InferSent(GPU)	21.24	0.15
SkipThought (GPU)	43.36	0.10

Table 5: Run time of GEM, InferSent and SkipThought on encoding sentences in STSB test set. GEM is run on CPU, InferSent and SkipThought is run on GPU. Data are collected from 5 trials.

- Much faster than parameterized models!

Results: sensitivity to hyper-parameters



Summary

- Proposed GEM parameter-free sentence encoder
  - ✓Mathematically-well founded
  - ✓Simple
  - ✓Fast inference
  - ✓Outperforms previous parameter-free models (sometimes comparable to parametrized one)
- Implementation is available at: https://github.com/fursovia/geometric_embedding