Word Embedding & CNN in NLP

by Jiawei

How do we represent the meaning of a word?

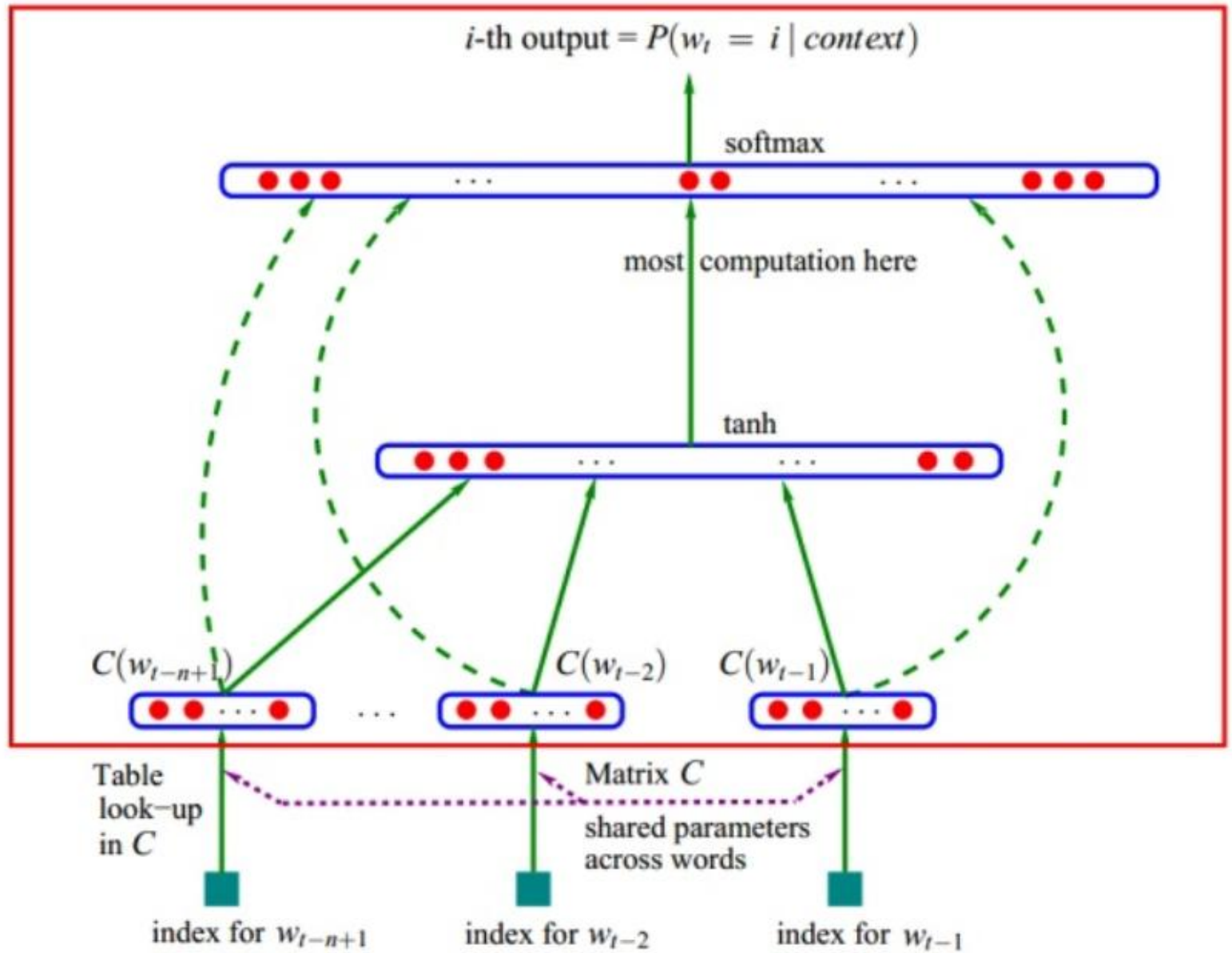
We call this a "one-hot" representation. Its problem:

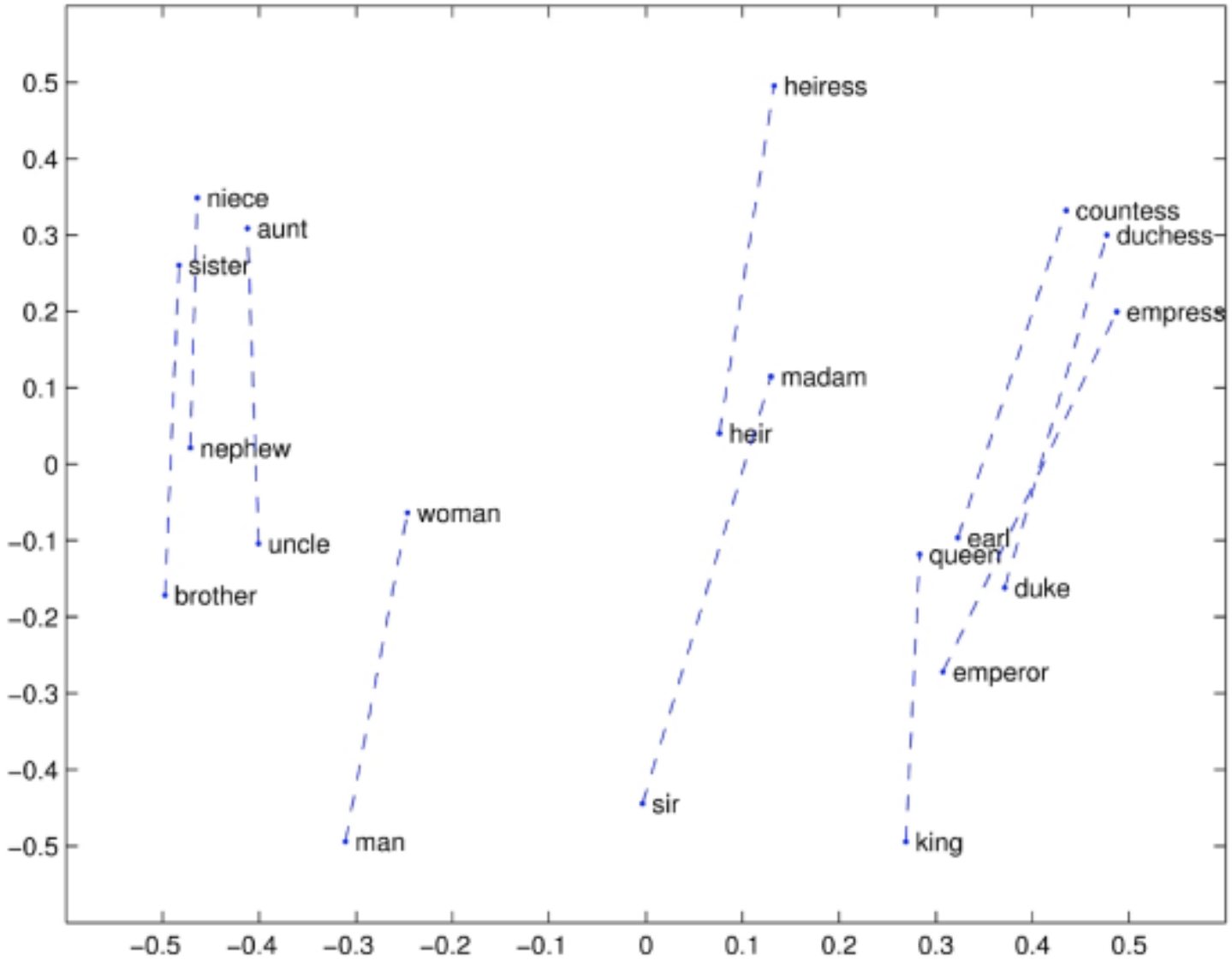
motel [0 0 0 0 0 0 0 0 0 0 1 0 0 0 0] AND
hotel [0 0 0 0 0 0 0 1 0 0 0 0 0 0 0] = 0

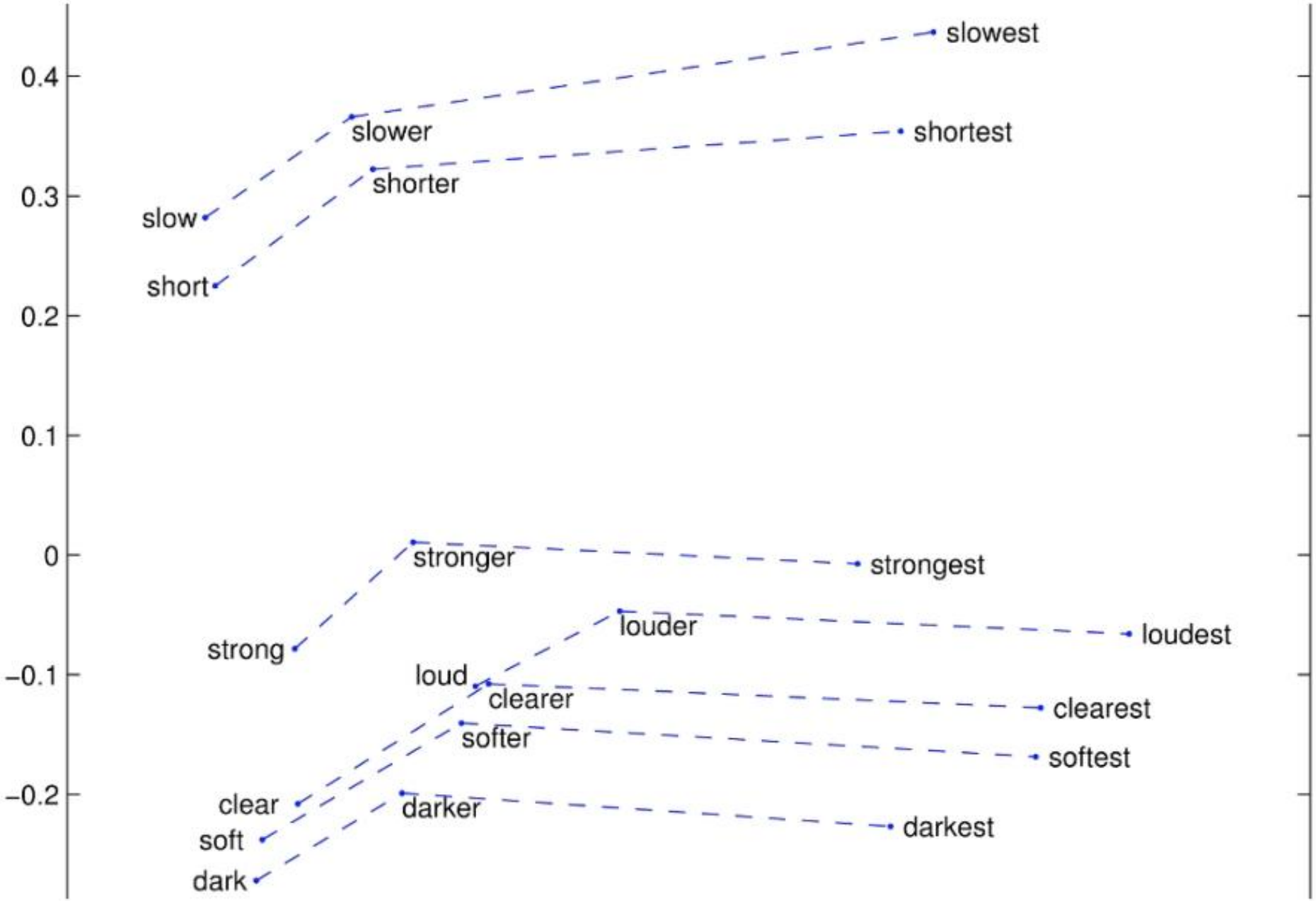
How do we represent the meaning of a word?

- How to make neighbors represent words?

- Use word2Vec


i-th output = $P(w_t = i \mid context)$
softmax
most computation here
tanh
$C(w_{t-n+1})$
$C(w_{t-2})$
$C(w_{t-1})$
Table look−up in C
Matrix C
shared parameters across words
index for w_{t-n+1}
index for w_{t-2}
index for w_{t-1}


0.5
0.4
0.3
0.2
0.1
0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.5
−0.4
−0.3
−0.2
−0.1
0
0.1
0.2
0.3
0.4
0.5
niece
aunt
sister
nephew
uncle
brother
woman
man
heiress
madam
heir
sir
countess
duchess
empress
earl
queen
duke
emperor
king


0.4
0.3
0.2
0.1
0
−0.1
−0.2
slow
slower
slowest
short
shorter
shortest
stronger
strongest
strong
louder
loudest
loud
clearer
clearest
softer
softest
clear
darker
darkest
soft
dark

Convolutional Neural Networks for Sentence Classification

- Yoon Kim New York University yhk255@nyu.edu

Abstract

- We report on a series of experiments with convolutional neural networks (CNN) trained on top of pre-trained word vectors for sentence-level classification tasks. We show that a simple CNN with little hyperparameter tuning and static vectors achieves excellent results on multiple benchmarks. Learning task-specific vectors through fine-tuning offers further gains in performance. We additionally propose a simple modification to the architecture to allow for the use of both task-specific and static vectors. The CNN models discussed herein improve upon the state of the art on 4 out of 7 tasks, which include sentiment analysis and question classification.

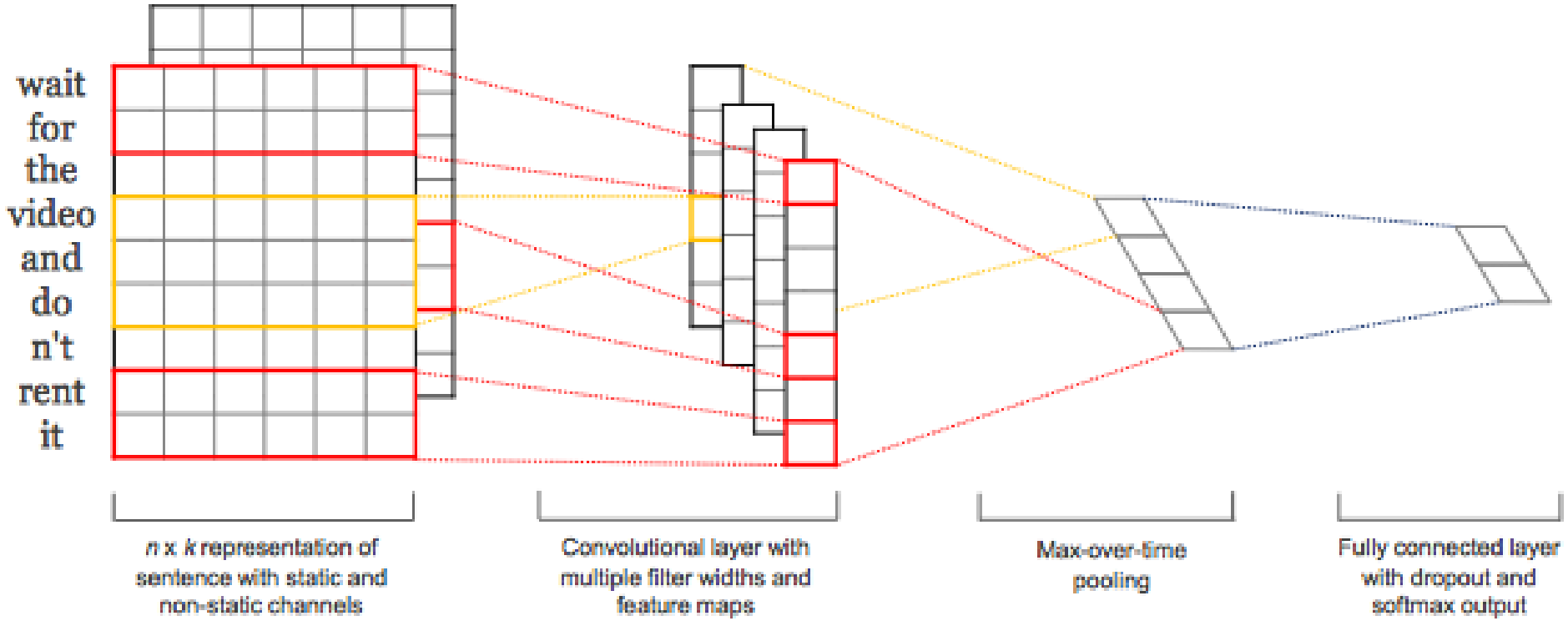


Figure 1: Model architecture with two channels for an example sentence.

Regularization

- Dropout

- We additionally constrain l 2 -norms of the weight vectors by rescaling w to have ||w|| 2 = s whenever ||w|| 2 > s after a gradient descent step.

Experimental Setup

- Pre-trained Word Vectors
- Initializing word vectors with those obtained from an unsupervised neural language model is a popular method to improve performance in the absence of a large supervised training set (Collobert et al., 2011; Socher et al., 2011; Iyyer et al., 2014). We use the publicly available word2vec vectors that were trained on 100 billion words from Google News. The vectors have dimensionality of 300 and were trained using the continuous bag-of-words architecture (Mikolov et al., 2013). Words not present in the set of pre-trained words are initialized randomly.

Model	MR	SST-1	SST-2	Subj	TREC	CR	MPQA
CNN-rand	76.1	45.0	82.7	89.6	91.2	79.8	83.4
CNN-static	81.0	45.5	86.8	93.0	92.8	84.7	**89.6**
CNN-non-static	**81.5**	48.0	87.2	93.4	93.6	84.3	89.5
CNN-multichannel	81.1	47.4	**88.1**	93.2	92.2	**85.0**	89.4
RAE (Socher et al., 2011)	77.7	43.2	82.4	–	–	–	86.4
MV-RNN (Socher et al., 2012)	79.0	44.4	82.9	–	–	–	–
RNTN (Socher et al., 2013)	–	45.7	85.4	–	–	–	–
DCNN (Kalchbrenner et al., 2014)	–	48.5	86.8	–	93.0	–	–
Paragraph-Vec (Le and Mikolov, 2014)	–	**48.7**	87.8	–	–	–	–
CCAE (Hermann and Blunsom, 2013)	77.8	–	–	–	–	–	87.2
Sent-Parser (Dong et al., 2014)	79.5	–	–	–	–	–	86.3
NBSVM (Wang and Manning, 2012)	79.4	–	–	93.2	–	81.8	86.3
MNB (Wang and Manning, 2012)	79.0	–	–	**93.6**	–	80.0	86.3
G-Dropout (Wang and Manning, 2013)	79.0	–	–	93.4	–	82.1	86.1
F-Dropout (Wang and Manning, 2013)	79.1	–	–	**93.6**	–	81.9	86.3
Tree-CRF (Nakagawa et al., 2010)	77.3	–	–	–	–	81.4	86.1
CRF-PR (Yang and Cardie, 2014)	–	–	–	–	–	82.7	–
SVM_S (Silva et al., 2011)	–	–	–	–	**95.0**	–	–

Results and Discussion

- Multichannel vs. Single Channel Models
- We had initially hoped that the multichannel architecture would prevent overfitting (by ensuring that the learned vectors do not deviate too far from the original values) and thus work better than the single channel model, especially on smaller datasets. The results, however, are mixed, and further work on regularizing the fine-tuning process is warranted. For instance, instead of using an additional channel for the non-static portion, one could maintain a single channel but employ extra dimensions that are allowed to be modified during training.

Static vs. Non-static Representations

- As is the case with the single channel non-static model, the multichannel model is able to fine-tune the non-static channel to make it more specific to the task-at-hand. For example, good is most similar to bad in word2vec, presumably because they are (almost) syntactically equivalent. But for vectors in the non-static channel that were fine-tuned on the SST-2 dataset, this is no longer the case (table 3). Similarly, good is arguably closer to nice than it is to great for expressing sentiment, and this is indeed reflected in the learned vectors.For (randomly initialized) tokens not in the set of pre-trained vectors, fine-tuning allows them to learn more meaningful representations: the network learns that exclamation marks are associated with effusive expressions and that commas are conjunctive (table 3).

Conclusion

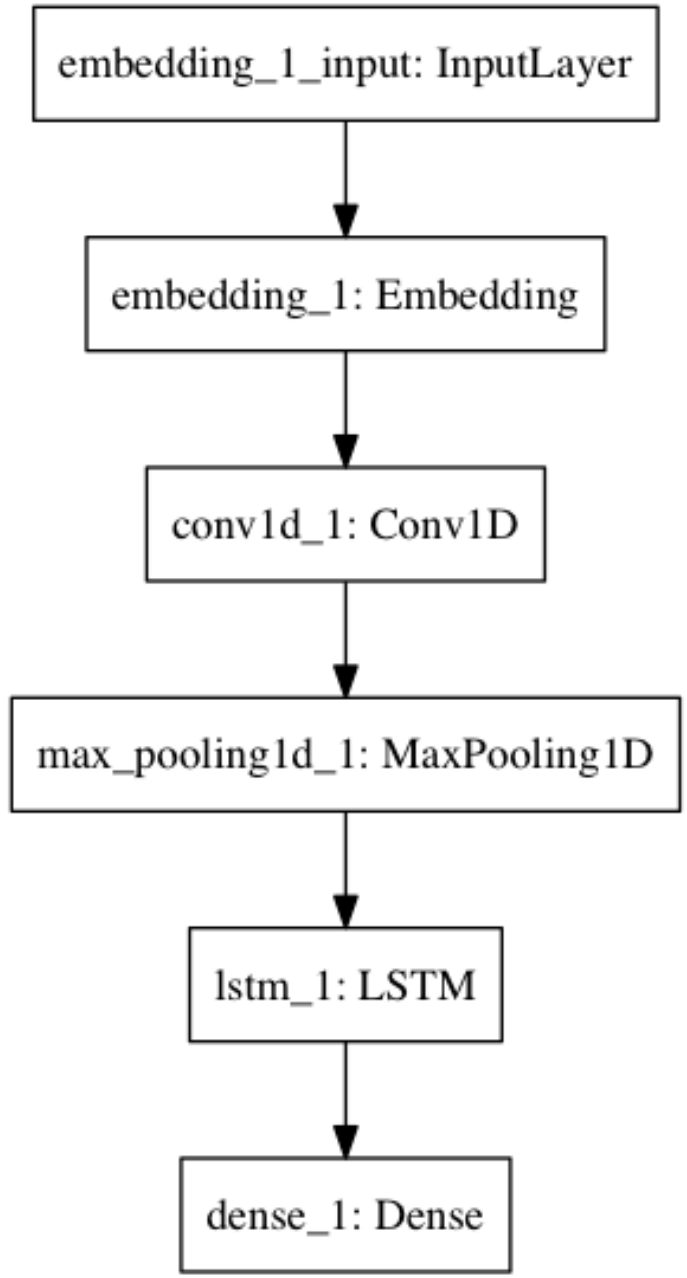
- In the present work we have described a series of experiments with convolutional neural networks built on top of word2vec. Despite little tuning of hyperparameters, a simple CNN with one layer of convolution performs remarkably well. Our results add to the well-established evidence that unsupervised pre-training of word vectors is an important ingredient in deep learning for NLP.

Recent Work

基于长文本新闻的情感分析

采用方法

- Word2Vec进行词向量pre-training
- 嵌套入CNN + RNN 的模型中进行训练
- Dropout


embedding_1_input: InputLayer
embedding_1: Embedding
conv1d_1: Conv1D
max_pooling1d_1: MaxPooling1D
lstm_1: LSTM
dense_1: Dense

难点

- 中文新闻情感数据集太少
- 新闻不同于微博，评论等主观性文本，具有很强的客观性，词语表述上不会像主观性文本那样含有强烈情感色彩的词语
- 新闻属于长文本类型，不同于普遍的段文本或者是句子文本，在做Sentiment analysis 时要做一定的改进

解决方案

- 采用中英翻译的方式，采用中文与英文间的词向量映射方式进行翻译。

- 将Word2Vec改进为Word2Doc，在句子或段的基础上构建词向量

References

- CS224d-Lecture by Richard Socher
- Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.
- Kim Y. Convolutional neural networks for sentence classification[J]. arXiv preprint arXiv:1408.5882, 2014.

Thanks