EECS 595 Natural Language Processing
Sentence Classification for News Titles

Chien-Wei Lin, I-An Huang, Hung-Wei Wu

Project Description

Classifying the semantic content is one of the critical problems in natural language processing. There are many cases where only a small number of words are provided to interpret the meaning or intent such as keyword searches. However, the performance of short text classification is limited due to shortness of sentences, which causes sparse vector representations if we use word occurrence to represent sentences, and lack of context. On the other hand, news titles, though consisting of short sentences, provide rich information of the semantic content in a concise way. Because of this property, we believe that news title classification will be a good start point for our sentence classification task.

With news classification in mind, we start with a representative dataset News Aggregator Data Set. In this way, we are free from the hassle of data preprocessing annotating date, removing undesired value, and compensating for missing fields of a sample. Besides, it's easier to find related work that also evaluates on this dataset, which makes comparison with our model easier or possible. Moreover, since there are only 4 news categories in this dataset, the task should be relatively easy and makes a great starting point for our project because of the potential difficulty described in the previous paragraph.

Methodology

Sentence Representation: Bag of Words vs. Word2Vec

Model

- Logistic Regression and SVM
- Convolution Neural Network (CNN)
  - Apply convolution on multiple windows with different length
  - Max pooling over time Capture the most important feature, which is the highest value in the feature map. Additionally, max pooling also helps us to down sample the features and handle the various filter lengths.

Regularization: Dropout

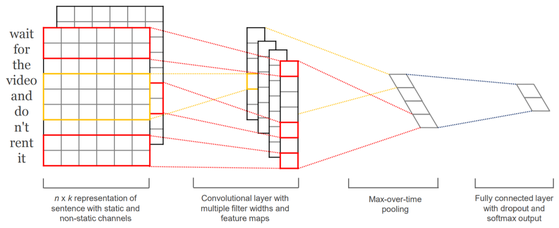


Figure.1 Model Structure

Sentence Representation

For every sentence s with length n, and $x_i \in \boldsymbol{R}^k$ be the k-dimensional word vector for $i = 1,2,\cdots n$. we can represent the sentence as $x_{1:n} = x_1 \oplus x_2 \oplus \cdots \oplus x_n$, where $\oplus$ is the concatenation operator. Let $x_{i:i+j}$ refer to the concatenation of words $x_i, x_{i+1} \cdots x_i$. A 1-D convolution involves a filter $w \in \boldsymbol{R}^{hk}$, which is applied to a window of h words to produce a new feature c_i, where

$$c_i = f(w \cdot x_{i:i+h-1} + b)$$

This filter is applied to each possible window of words $\{x_{1:h}, x_{2:h+1}, \dots, x_{n-h+1:n}\}$ to produce a feature map

$$\boldsymbol{c} = [c_1, c_2, \dots, c_{n-h+1}]$$

Finally, we apply $\boldsymbol{c}$ to the max-pooling component $\hat{c} = \max \boldsymbol{c}$ as the feature corresponding to this particular filter.

Conclusion

From the tables, it can be shown that CNN with BoW has a quite poor performance. However, it's wrong to conclude that CNN is an inferior NLP model because its performance is comparable to the other two using w2v. We consider it to be because BoW model does not align with the nature of convolution and the net discards too much information using a max pooling strategy.

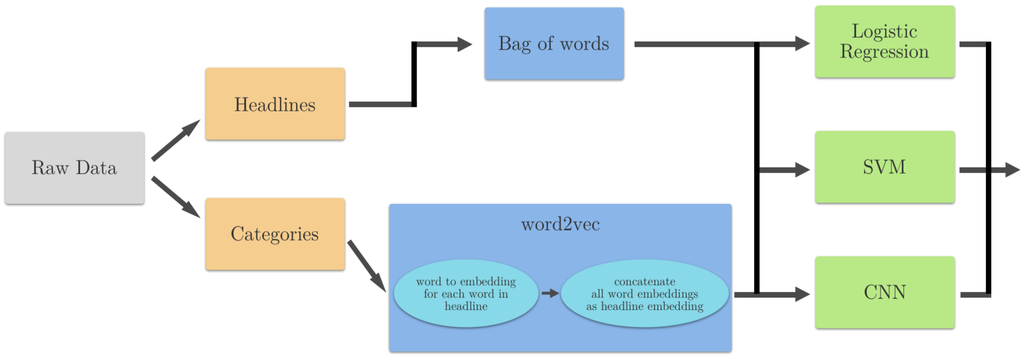
Regarding embedding, we observe that word2vec embedding is quite robust against the size of dataset and this is probably because it contains information about related words. Besides, CNN tends to perform better using word2vec than the other two models. We attribute this to the time-invariant property of convolution. On the other hand, we also notice that if we do not introduce a reduction in the dimension of BoW vectors, BoW actually perform significantly better than word2vec when combined with SVM and logistic regression. This might imply that for this kind of task word frequency is much more useful than the meaning of words.

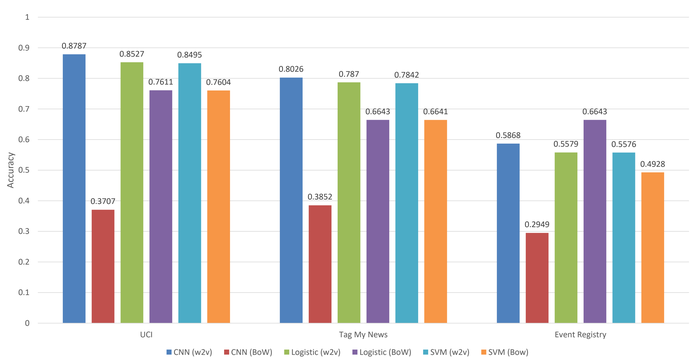
As for dataset, it can be seen that on News Aggregator models are performing significantly better than on the other two. This is possibly due to the nature of datasets. For News Aggregator, many words have very high frequency in a given category. For example, Google occurs over 10% in technology category and Apple occurs roughly 10%. Intuitively, this will make classification easier, especially when we use full dimensionality of BoW vectors.

Future Direction

We want to experiment with skip-thought vectors as they are generated by considering sentences as a whole and have generally good performance on many tasks.

Furthermore, fastText is known to be good at syntactic tasks. It might be interesting to compare the resulting embedding against those from word2vec.





Dataset

	Tokens	Instances	Vocabulary	Category
News Aggregator (NA)	3.6B	69K	64K	4
Tag My News (TMN)	313M	17K	24K	5
Event Registry (ER)	3B	54K	82K	4

Metrics

NA/TMN/ ER	CNN W2V	CNN BoW	Logistic W2V	Logistic BoW	SVM W2V	SVM BoW
Precision	0.878/0.8/ 0.585	0.23/0.148/ 0.275	0.844/0.784/ 0.555	0.76/0.667/ 0.492	0.849/0.782/ 0.555	0.76/0.666/ 0.495
Recall	0.879/0.803/ 0.587	0.371/0.385/ 0.287	0.845/0.787/ 0.558	0.761/0.664/ 0.49	0.85/0.784/ 0.58	0.76/0.664/ 0.493
F1	0.878/0.8/ 0.585	0.221/0.214/ 0.227	0.844/0.784/ 0.555	0.759/0.65/ 0.49	0.849/0.782/ 0.555	0.758/0.65/ 0.493

Reference

- Zhao, R.; Mao, K. Topic-Aware Deep Compositional Models for Sentence Classification. IEEE/ACM Trans. Audio Speech Lang. Process. 2017, 25, 248–260.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013.
- Kim, Yoon. 2014. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 .
- Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. "Skip-Thought Vectors." arXiv preprint arXiv:1506.06726 (2015).