Phrase2Vec: Identifying Phrases and Learning Their Vector Embeddings

Jiali Lu
jialilu.physics@gmail.com

1. Introduction

Vector embeddings, such as Word2Vec [1] and GloVe [2], provides fast and efficient solutions to many Natural Language Processing (NLP) tasks. The main limitation of this approach is that it cannot account for idiomatic expressions or negation. In this project, I resolve the limitation by identifying phrases and learn their embeddings. The embedding groups semantically similar phrases together, and shows linear substructure just as in the word embedding case. When applied to sentiment analysis task, using phrase embedding improves accuracy by 2% compared to word embedding.

2. Method

2.1. Identifying Phrases

Learning embeddings of phrases has been considered in the original Word2Vec paper [1]. However, their method fails beyond bigrams.

In this report I improve upon their method and propose a new way to identify phrases based on association. For each N-gram $x = (x_1, x_2, ..., x_N)$ with length N and relative frequency $n(x) > s$, I compute a score

$$\text{score}(x) = \min_k \frac{n(x)}{n(x_1^k) \times n(x_{k+1}^N)}$$

and those with scores higher than a threshold t is designated as phrases and added to the vocabulary. Here s is the frequency threshold and $x_a^b = (x_a, x_{a+1}, ..., x_b)$ stands for a segment of s with index between a and b. The parameters s and t are to be chosen by the user. In this report I chose $s = 1 \times 10^{-6}$. As for t, I use $t = 100$ when demonstrating semantic similarity and linear substructure, but use $t = 10$ for text classification. The main reason is that lower threshold (like $t = 10$) learns loose phrases like ”not acceptable”. While such phrases are valuable for sentiment analysis, they are distracting for human interpretation.

2.2. Learning Phrase Embeddings

Phrase Embedding can then be learned similar to word embedding. In this case I chose continuous bag-of-word (CBOW) formalism [1].

The only difficulty in tokenizing text into words and phrases of variable word length. This can be efficiently done by using a prefix trie, where each path corresponds to a phrase (or word), and the next token can be found by looking for the longest common path in the trie.

TABLE 1. PHRASE SIMILARITY, FOOD AND DRINK

Query Phrase	Most Similar Phrases (top 3)
kung pao chicken	mongolian beef, orange chicken, sesame chicken
lamb vindaloo	chicken tikka masala, saag paneer, lamb curry
red snapper	snapper, halibut, sea bass
Sam Adams	Fat Tire, Yuengling, Stella
Pinot Noir	Cabernet, Sauvignon Blanc, Malbec

3. Result

For this project I use Yelp review dataset [3]. The phrases are learned from the 650k reviews in training set, while the accuracy for sentiment analysis is calculated from the test set. Yelp review dataset contains information on many restaurants and stores, offering us interesting interpretation for the phrase embeddings.

3.1. Phrase Similarity

As in the original Word2Vec paper [1], our phrase embedding groups semantically similar phrases together in the vector space. This is shown in Table 1-4. Note the phrase embedding is able to appreciate idioms, such as ”knock your socks off” and negation, as in ”do not eat here”.

3.2. Linear Substructure

In the GloVe paper [2], it is observed that the vectors mapping companies to executives are almost parallel, and this is called linear substructure. Similar observation can be made in Yelp reviews, between regions and their signature dish. This is visualized using PCA in Fig 1.

TABLE 2. PHRASE SIMILARITY, LOCATIONS

Query Phrase	Most Similar Phrases (top 3)
gas station	Circle K, oconvenience store, 7-11
Trader Joe's	Whole Foods, Trader Joes, Safeway
PF Changs	Panda Express, PF Chang's, Pei Wei
Red Lobster	Olive Garden, Outback, Applebees
Hard Rock	Hard Rock Hotel, Palms, Cosmopolitan

TABLE 3. PHRASE SIMILARITY, SENTIMENT

Query Phrase	Most Similar Phrases (top 3)
stay away	avoid this place at all costs, do not eat here, do not stay here
come back	return, go there again, eat here again
false advertisement	false advertising, bullshit, scam
blow you away	blow your mind, knock your socks off, bring me back
waste of money	waste of time and money, waste of time, ripoff

TABLE 4. PHRASE SIMILARITY, MISCELLANEOUS

Query Phrase	Most Similar Phrases (top 3)
American Express	Amex, Visa, debit cards
Black Friday	Labor Day, holiday, opening weekend
buy one get one free	BOGO, 2 for 1, 2-for-1
Michael Jackson	Britney Spears, Celine Dion, Elvis
Star Wars	Harry Potter, Pawn Stars, Memorabilia

3.3. Text Classification: Sentiment Analysis

In addition to nice interpretation, phrase embedding also increases model accuracy compared to simple word embedding. To illustrate this, I perform sentiment analysis based on average phrase and word embedding feature using gradient boosting machine. The accuracies are listed in Table 5.

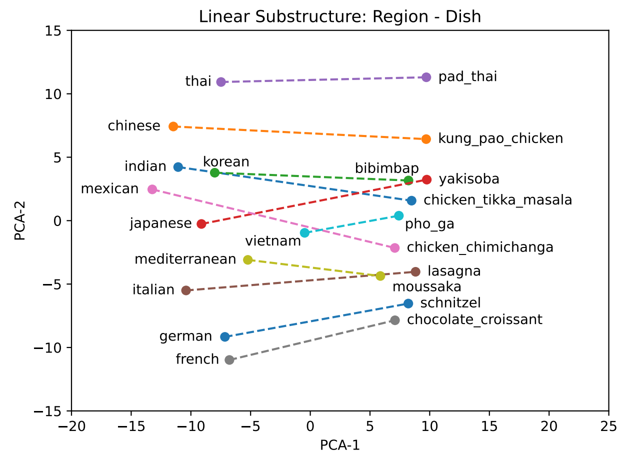

Figure 1. Linear Substructure. The vectors mapping regions to their signature dishes are overall aligned.

TABLE 5. SENTIMENT ANALYSIS ACCURACY

	accuracy (5 class)	accuracy (polarity)
word embedding	0.564	0.919
phrase embedding	0.588	0.941

As we can see, using phrase embedding increase accuracy by more than 2% both when measured in class and polarity. The accuracy could be further improved using more tricks [4].

4. Conclusion

In this project I identified phrases from Yelp reviews and learned their vector embeddings. The embedding has nice interpretation. It groups semantically similar phrases together and have linear substructure within region-dish pairs. In addition, when used in place of word embeddings, phrase embedding increases sentiment analysis accuracy by more than 2%. Phrase embedding could be a fast and efficient tool for many NLP tasks.

References

[1] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, 26.

[2] Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).

[3] Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. Advances in neural information processing systems, 28.

[4] Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification. arXiv preprint arXiv:1607.01759.