Document Embedding for Scientific Articles: Validation of word embeddings

Arjan Meijer, 11425555

April 9, 2018

1 Abstract

[TODO: Abstract]

Document Embedding

The basic theory of word embedding can be explained by the following quote of J.R. Firth: "You shall know a word by the company it keeps". Word embeddings are distributed representations of words (Mikolov et al. [1]), which can capture both the semantic and syntactic information of words from a large unlabeled corpus (Lai et al. [2]). Word embeddings are vector based representations of words, that can, depending on the model either predict the target word given context words, or predict context words, given the target word. Techniques based on word vectors have improved various NLP areas such as named entity recognition, part-of-speech tagging, parsing, and semantic role labelling (Luong et al. [3]). The word2vec model converts words via a learned lookuptable into real valued vectors [4]. Mikolov et al. [4] show that calculations with these vectors is also possible:

$$\mathrm{X}_{\mathrm{apple}} - \mathrm{X}_{\mathrm{apples}} \approx \mathrm{X}_{\mathrm{car}} - \mathrm{X}_{\mathrm{cars}}$$

Furthermore they show that de distance in the vector space between "king" and "man" approximates the distance between "queen" and "women". Variations on the word2vec model have also been proposed, Le and Mikolov [5] introduced the paragraph vector, based on the word2vec model. The paragraph vector model uses additional variables to improve the accuracy of the word-embeddings. An advantage of the paragraph vector model is that it takes the word order into consideration, atleast in a small context [5]. For this reaseach, only word2vec will be used.

Domain specific

Earlier work on this topic, concerning academic articles, by (Truong [6]) states the following:

> "Our findings clearly evinced that in-domain training of the word embeddings can drastically improve the process of document clustering. In fact, the effect is even stronger than the number of training examples and the model architecture. However, too isolated training can lead to a failure of several clustering algorithms, such as DBSCAN, due to a too dense vocabulary". (Truong [6])

This was also found by Lai et al, who state:

> "The corpus domain is more important than the corpus size. Using an in-domain corpus significantly improves the performance for a given task, whereas using a corpus in an unsuitable domain may decrease performance" (Lai et al. [2]).

These statements both indicate that an in-domain corpus improves the performance of word-vectors for those specific domains. However, domain specific validation techniques do not exist currently. Multiple generic validation sets are publicly available such as: the Rare-word dataset introduced in the paper "Better Word Representations with Recursive Neural Networks for Morphology" (Luong et al. [3]), the MEN test collection (Bruni [7]) and the WordSimilarity-353 test collection (Gabrilovich [8]). These sets have been used in multiple studies of word-vectors, whom are referenced by the respective sources. These validation methods are limited to the non domain specific texts, since they do not contain words of specific domains.

Embedding validation

Not published results of the study by Truong show high error rates on the validation scores, this is presented in Table 1. However, the word-vectors work well on document clustering, this seems to indicate that the word-vectors are able to represent the words. The problem is then that the currently used validation methods do not confirm or this. Therefore, we propose the validation of word-embeddings through [TODO: SOLUTION - classification - abstract/text/title matching - keyword categorization - ...].

	WordSim	Men	RareWords
Lowest	0.38	0.54	0.29
Highest	0.49	0.61	0.32
Average	0.45	0.59	0.32
Baseline	0.64	0.68	0.34

Table 1: Table 1

References

[1] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[2] Siwei Lai, Kang Liu, Shizhu He, and Jun Zhao. How to generate a good word embedding. *IEEE Intelligent Systems*, 31(6):5–14, 2016.

[3] Thang Luong, Richard Socher, and Christopher Manning. Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113, 2013.

[4] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, 2013.

[5] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196, 2014.

[6] J. Truong. An evaluation of the word mover's distance and the centroid method in the problem of document clustering, 2017.

[7] Elia Bruni. Men test collection, 2012. URL `https://staff.fnwi.uva.nl/e.bruni/MEN`.

[8] Evgeniy Gabrilovich. Wordsimilarity-353 test collection, 2002. URL `http://www.cs.technion.ac.il/ gabr/resources/data/wordsim353/`.