[Paper Review]
Enriching Word Vectors with Subword Information

Paul Jason Mello
Department of Computer Science and Engineering
University of Nevada, Reno
pmello@unr.edu

Abstract

"Continuous word representations, trained on large unlabeled corpora are useful for many natural language processing tasks. Popular models that learn such representations ignore the morphology of words, by assigning a distinct vector to each word. This is a limitation, especially for languages with large vocabularies and many rare words. In this paper, we propose a new approach based on the skip-gram model, where each word is represented as a bag of character n-grams. A vector representation is associated to each character n-gram; words being represented as the sum of these representations. Our method is fast, allowing to train models on large corpora quickly and allows us to compute word representations for words that did not appear in the training data. We evaluate our word representations on nine different languages, both on word similarity and analogy tasks. By comparing to recently proposed morphological word representations, we show that our vectors achieve state-of-the-art performance on these tasks." [1]

1 Summary

Recognizing that some languages have complex structures and finite units of meaning at the character level (morpheme), the authors of this paper dare to ask the following question. What if we trained an n-gram model on better data?

2 Main Contributions

The main contributions of this paper revolve around a few key features as described below.

2.1 Key Contributions

- The authors represent words in a vector format by decomposing each word into a continuous bag of character level n-grams resulting in each word becoming the sum of n-gram vectors. For example, when $n = 3$, the word "*where*" is broken down into `<whe>`, `<her>`, and `<ere>`, with each n-gram representation containing the complete word itself, resulting in the set [`<whe>`, `<her>`, `<ere>`, `<where>`].
- Through decomposition of words into n-gram characters, they do not need implicit morphological decomposition of words as other papers utilize.

2.2 Innovative Aspects

The most innovative aspects have to be two core ideas:

- The understanding that across all languages, words and sentences contain intrinsic and relational meaning which can be captured through character level vector representations.
- Rather than representing these words as a single, complete, and unique vector representation, they take the sum of each character n-gram and vectorize the data.

3 Strengths and Weaknesses

While this paper is an improvement on prior SOTA approaches, there remain weaknesses and future areas of improvement.

3.1 Strengths

This paper builds off the previous work of word2vec[2] making the implementation easily accessible, simple to use, and powerful. The decomposition of words into subword n-grams improves the generalization capabilities of the model to unseen data by breaking down the vector representations into a sum of finite characters, effectively "enriching" the vector space. Particular success is found in morphologically rich languages like Turkish and in situations where data is sparse or, in some cases, unseen. Their utilization of a novel score function which maps words to context also provides another level of improvements in training. Their approach is fast, reliable, and achieves SOTA results in every tested language.

3.2 Weaknesses

While their approach is simple and effective, better techniques to improve the data quality, particularly through pruning subword n-grams which do not contribute meaningful or relevant information to the vector representation would further improve the representations. The authors note that even as data is increased, their model becomes saturated and does not necessarily result in better training outcomes. Their approach also requires considerable memory and speed optimizations in order to only achieve a 1.5x slowdown in training when compared to the baseline Skipgram model.

3.3 Areas of Improvements

As previously mentioned, one potential point of improvement would be to prune the subword n-grams to remove subwords which contain little to no information. Another obvious but overlooked improvement could be to utilizing a dynamic approach to the length of n-grams even within each word to effectively map representations.

4 Discussion

While this paper is a step in the right direction its simplicity is a double edged sword. On one hand, by capturing all the n-gram subword data, it is a highly effective and simple way to generate enriched vector representations through the decomposition of words. On the other hand it completely ignores the noisy subword n-grams generated within each subword n-gram. Effectively, one could make a significantly better model, with far richer vector representations, by incorporating better semantic decompositions of each word through their root, synonyms, antonyms, gender, and tense. However, for the sake of simplicity, pruning irrelevant or noisy subwords in these n-grams would significantly the model.

5 Conclusion

This paper represents a large leap in natural language processing with its contributions of simple tokenization and embedding techniques for 2016. It's decomposition of words into subword n-grams proves effective in mapping and generating understanding due to the increase in data for each vector. Rather than utilizing a unique vector, its sum of n-gram vectors approach trains a robust and effective model capable of generalize beyond the training data. Although, I will add that this claim is likely a misunderstanding as those subword n-grams must contain the information for unseen data, which appears to generalize beyond the training data. Today these approaches have been largely

forgotten in favor of in context learners such as the transformer architecture [3] proposed in the following year of 2017.

References

[1] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information, 2017.

[2] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.