The impact of tokenizers on word embeddings: Analysis of similar words and WEAT score

Anonymous Author(s)
Affiliation
Address
email

Abstract

This study investigates the impact of tokenizer selection on word embeddings, particularly in relation to biases and the quality of similar word predictions. While prior research has primarily focused on comparing specific embedding models, such as Word2Vec and FastText, systematic analyses of how tokenizers influence word embeddings are scarce. Using movie synopsis data tokenized with the Okt tokenizer, we applied Word2Vec and FastText to measure association using the Word Embedding Association Test (WEAT) score. The results revealed that FastText performed worse than Word2Vec in terms of qualitative evaluation based on similar word lists, and exhibited stronger associations, which challenges the assumption that FastText is better suited for Korean embeddings. We hypothesized that performance of FastText could be improved with a subword-based tokenizer and explored this in our experiments. The findings demonstrate that the combination of tokenizer and embedding model can significantly affect qualities of word embeddings and association, influencing metrics like the WEAT score.

1 Introduction

Word embedding models are essential components in natural language processing (NLP), and extensive research has been conducted to analyze their inherent biases. In particular, studies utilizing the Word Embedding Association Test (WEAT) score have contributed to evaluating social biases in word embeddings. However, most prior research has primarily focused on analyzing differences between specific embedding models (e.g., Word2Vec, FastText), while ****systematic studies on the impact of tokenizer selection on word embeddings remain limited.

In the Going Deeper Node 3 project, we conducted experiments to measure bias in word embeddings. Specifically, we tokenized movie synopsis data created between 2001 and August 2019 using the Okt tokenizer and then embedded the text using Word2Vec and FastText. Subsequently, we defined two target word sets (regular movies vs art movies) and an attribute set, computed the WEAT score, and compared the lists of similar words for given query terms.

The experimental results revealed that FastText performed worse than Word2Vec in qualitative evaluations based on similar word lists. For instance, when retrieving similar words for the query word "영화" (movie), Word2Vec produced relevant words such as "영화로" (to movie), "작품" (work), "코미디" (comedy), and "전작" (previous work). In contrast, FastText primarily returned compound words containing "영화" (movie), such as "영화로" (to movie), "극영화" (feature film), "인도영화" (Indian movie), and "청춘영화" (youth movie). Moreover, the bias measured by the WEAT score was significantly more pronounced in FastText compared to Word2Vec. These findings contradict the commonsense that FastText, which computes a subword-based embedding, is better suited for Korean word embeddings. Given these results, one might argue that FastText is not a more reasonable choice than Word2Vec.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

Based on these observations, we explored the possibility that the performance of FastText could vary depending on the tokenizer used. Unlike Word2Vec, which embeds words at the word-level, FastText learns subword representations, suggesting that applying FastText to text tokenized with a subword-based tokenizer could yield better results.

This study experimentally analyzes how the combination of tokenizer and embedding model affects bias in word embeddings and the quality of similar word predictions. Our findings demonstrate that even when using the same embedding model, differences in tokenization methods can influence both bias and expressiveness, impacting evaluation metrics such as the WEAT score.

2 Methods

2.1 WEAT score

The WEAT score is a method for quantitatively measuring association implied in word embeddings. To mathematically model the calculation of the WEAT score, we define an X-Y concept axis composed of two target word sets and an A-B concept axis composed of two attribute word sets. If there is no association in the target word sets along attribute axis, words in X and Y should have similar distances to words in A and B, respectively.

Based on this, the WEAT score is mathematically defined as follows:

$$S(X, Y, A, B) = \frac{mean_{x \in X} s(x, A, B) - mean_{y \in Y} s(y, A, B)}{std_{w \in X \cup Y} s(w, A, B)},$$

$$s(w, A, B) = \frac{1}{|A|} \sum_{a \in A} \cos(w, a) - \frac{1}{|B|} \sum_{b \in B} \cos(w, b).$$

$s(w, A, B)$ represents how much the similarity of a target word w differs between the two attribute sets A and B. This association value ranges from -2 to 2, and the larger its absolute value, the more biased w is along the A-B attribute axis.

To represent the difference between association of set X and Y, $S(X, Y, A, B)$, normalize the difference between the averages of the associations of the two sets by the standard deviation of the association values.

2.2 Experiments

In this experiment, two different tokenizers (Okt and Mecab) are used to tokenize text, and two embedding models (Word2Vec and FastText) are applied independently. This results in a total of four test cases. For these four test cases, we will examine the impact of tokenizers on the performance of word embedding models by analyzing both the retrieved similar words for query words and the WEAT score in the **Results** section.

3 Results

3.1 Similar words evaluation

Okt Tokenizer Figure 1 shows the retrieved similar words when applying Word2Vec and FastText to text tokenized with Okt. FastText, which embeds words using subword-based approach, is known to be more suitable for Korean word embeddings. However, when listing words closely related to "영화" (movie), we can see that Word2Vec retrieves words that are conceptually associated with movies, whereas FastText just lists different movie genres.

This phenomenon can be attributed to FastText's subword-based embedding computation. Since FastText calculates embeddings by considering subword units, when computing the embedding of a compound word containing "영화," the embedding of "영화" itself is heavily reflected. As a result, many compound words containing the query word tend to appear among the retrieved similar words. However, this issue could be mitigated by using a tokenizer that performs more fine-grained

Figure 1: Similar words of '시작' and '영화' in Okt - Word2Vec(left) combination and Okt - FastText(right) combination

tokenization. For example, when tokenizing the word "인도영화" (Indian movie) with Mecab, it is split into "인도" (India) and "영화" (movie), recognizing them as separate words. This would reduce the occurrence of compound words being retrieved as similar words.

Mecab Tokenizer Figure 2 shows the retrieved similar words when applying Word2Vec and FastText to text tokenized with Mecab. Since Mecab performs more precise morphological analysis based on detailed part-of-speech tagging compared to Okt, we observed that words more closely related to the query word were listed as similar words. In particular, when embedding the word "영화" (movie) using FastText, the retrieved similar words were significantly more reliable than those obtained with Okt tokenization. Additionally, for other words as well, FastText produced more reasonable similar words compared to Word2Vec.

These experimental results suggest that the choice of tokenizer can have a significant impact on the performance of word embedding models. In the following experiment, we will analyze the influence of tokenizers on word embeddings by examining the WEAT score for different tokenizer-embedding model combinations.

3.2 WEAT score

Figures 3 to 6 visualize the WEAT scores for the four test cases in the form of heatmaps. In the case of text tokenized with Okt (Figures 3 and 4), no significant differences were observed in the overall pattern of WEAT scores between the two word embedding methods. However, since FastText has slightly higher absolute WEAT scores than Word2Vec (the heatmap color for FastText is visibly darker), we can infer that FastText tends to highlight associations in word representations more than Word2Vec.

On the other hand, when performing association analysis on text tokenized with Mecab (Figures 5 and 6), the absolute difference between the heatmaps of Word2Vec and FastText was less pronounced than before. In fact, referring back to the cosine similarity results of the word relationships represented in section 3-1, we can see that after tokenizing with Okt and embedding the text, the results of the two

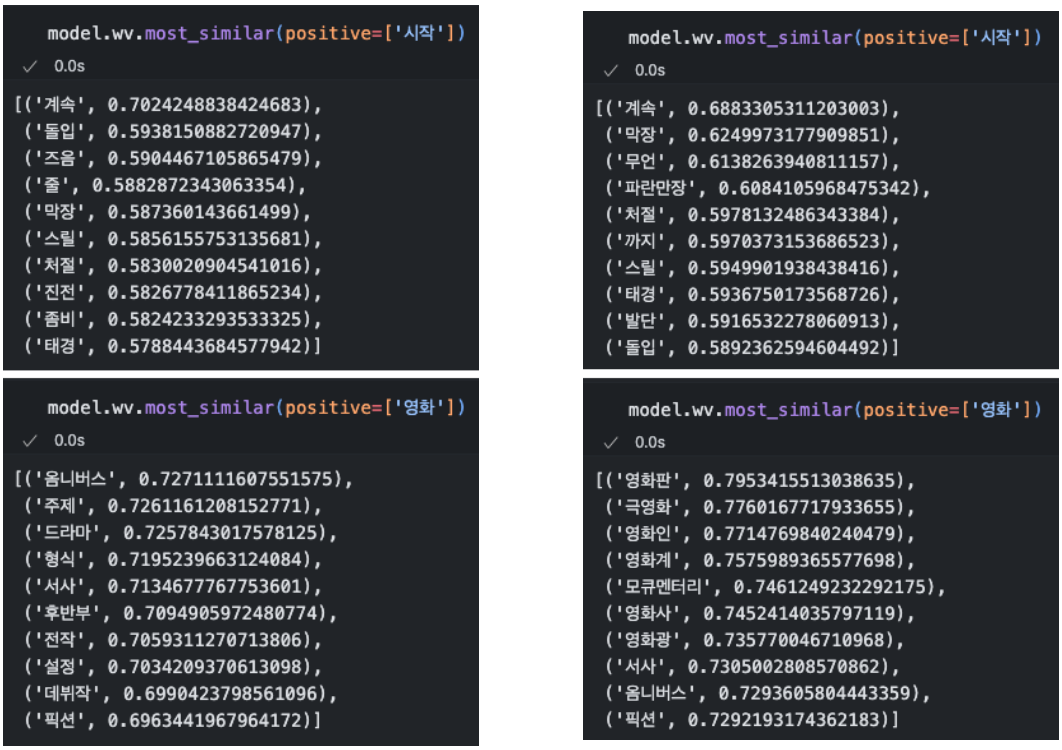

Figure 2: Similar words of '시작' and '영화' in Mecab - Word2Vec(left) combination and Mecab - FastText(right) combination

embeddings showed more distinct differences. However, when tokenizing with Mecab, this difference diminished, and this was reflected in the WEAT scores as well.

4 Conclusion

This study demonstrates that the combination of tokenizer and word embedding model significantly affects the association and performance of word embeddings, with a particular impact on the quality of similar word predictions and bias measurements using the WEAT score. Through our experiments, we observed that FastText, which uses subword embeddings, did not always outperform Word2Vec as expected, especially when tokenized with the Okt tokenizer. FastText's tendency to return compound words as similar words suggests that tokenization plays a crucial role in determining the effectiveness of subword-based embeddings.

Furthermore, we found that using a more precise tokenizer like Mecab improved the quality of similar word predictions, especially for FastText, which could better handle compound words by splitting them into their subword components. This indicates that tokenizer choice not only affects the semantic similarity of words but also influences the strength of associations detected by the WEAT score. Specifically, tokenization with Mecab resulted in less pronounced association in the embedding models, highlighting the importance of choosing appropriate tokenization strategies to mitigate the effect of tokenizer-dependent associations.

In conclusion, while FastText is often considered more suited for Korean word embeddings due to its subword-based approach, our results challenge this assumption by showing that tokenizer selection significantly impacts its performance. Tokenizers that provide more granular word-level segmentation, such as Mecab, can lead to better results with FastText, mitigating some of the associations and improving the quality of similar word retrieval. This study underscores the need for careful consideration of tokenization methods in word embedding tasks, particularly when measuring semantic similarity and associations in korean NLP applications.

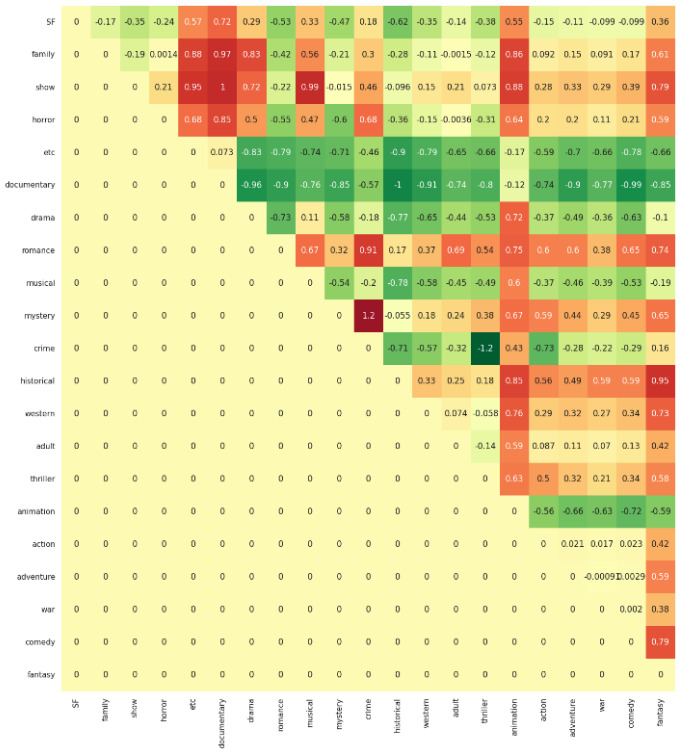

Figure 3: The result of WEAT score in Okt - Word2Vec combination

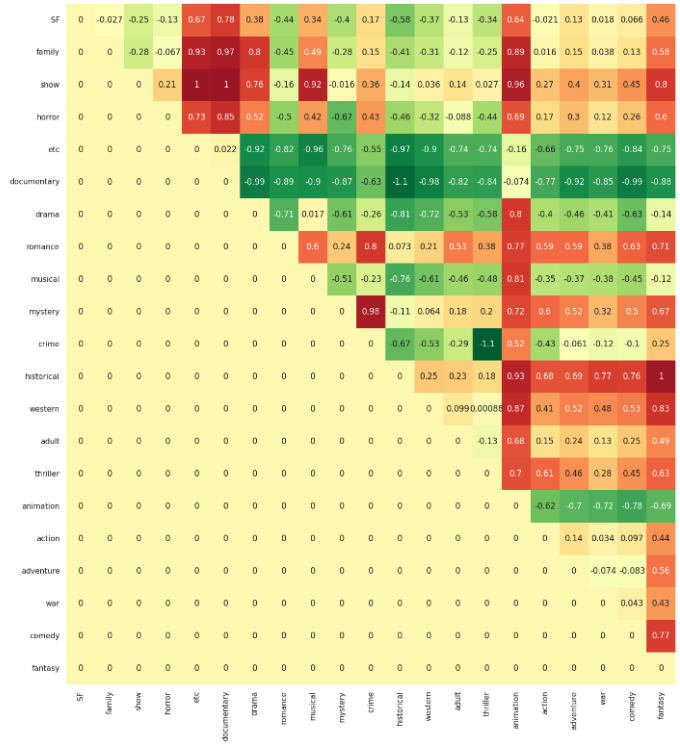

Figure 4: The result of WEAT score in Okt - FastText combination

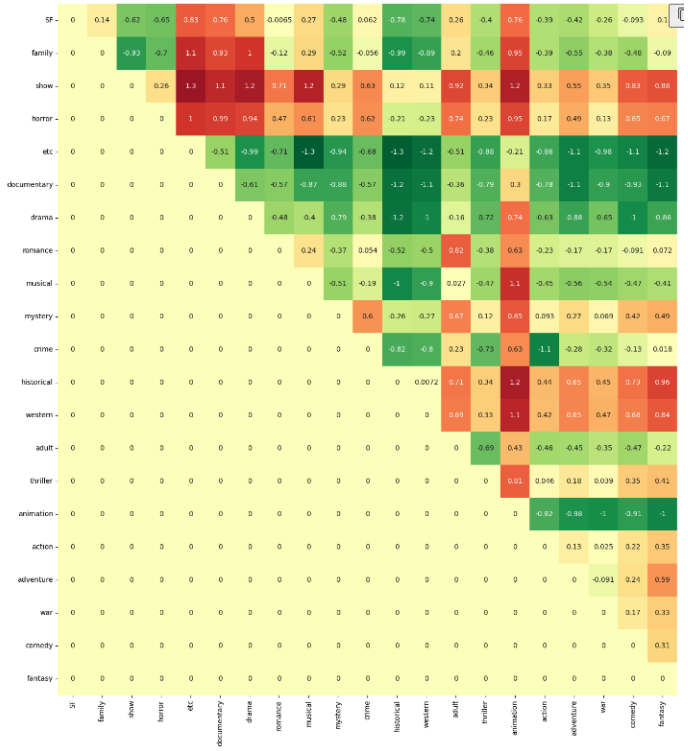

Figure 5: The result of WEAT score in Mecab - Word2Vec combination

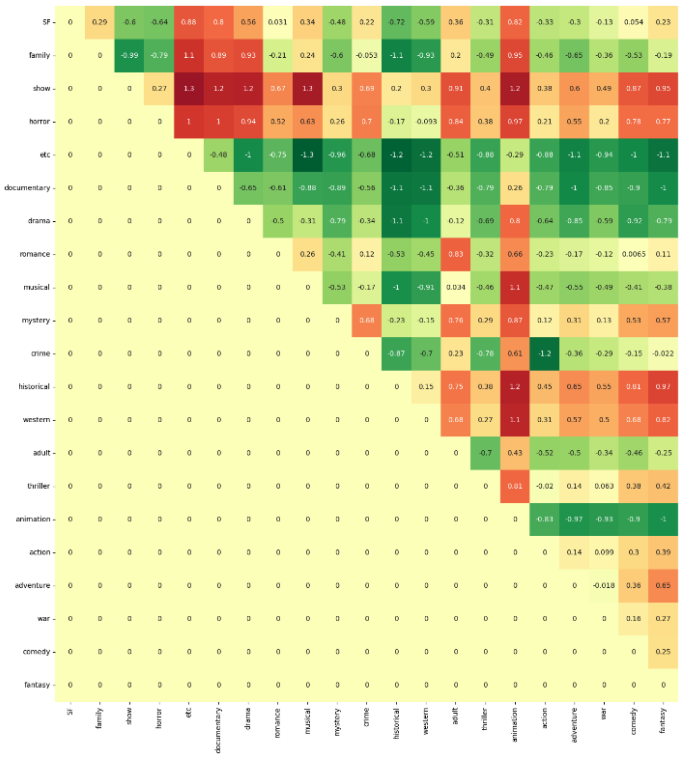

Figure 6: The result of WEAT score in Mecab - FastText combination

References

[1] Mikolov, T., Chen, K., Corrado, G. & Dean, J. (2013) Efficient estimation of word representations in vector space. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

[2] Bojanowski, P., Grave, E., Mikolov, T., Puhrsch, C. & Joulin, A. (2017) Enriching word vectors with subword information. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

[3] Caliskan, A., Bryson, J.J. & Narayanan, A. (2017) Semantics derived automatically from language corpora contain human-like biases. *Science* **356**(6334):183-186. DOI: 10.1126/science.aal4230.