Assessing Word Embedding Evaluation Methodologies

Cassandra Durr [1]

Abstract

The aim of this study is to describe a number of commonly used approaches for assessing the quality of learned word embeddings, and determine the efficacy of the evaluation criteria. While learning word embeddings is a vital first step for a number of natural language processing applications, the lack of a universally accepted evaluation criterion poses a significant challenge in assessing word embeddings' performance. We suggest semantic textual similarity and assessing the downstream performance of embeddings as generally sensible evaluation criteria.

1. Introduction

Learning appropriate word embeddings is crucial for a number of tasks in natural language processing, such as informational retrieval, machine translation, dependency parsing and question answering (Wang et al., 2019). However, despite the importance of learning quality word latent representations, there is currently no accepted gold-standard evaluation criteria for assessing word embeddings. The primary focus of this study is to describe a set of evaluation criteria for a word embedding model, along with their benefits and drawbacks.

2. Data Processing

The dataset used in this study is the AG's News Topic Classification Dataset which consists of 120,000 training instances and 7,600 test instances where an instance is a news headline from one of four categories: sport, world-news, business and science/technology. In this study, 15% of the training data is reserved as a validation set.

Data processing involved several steps: replacing singular digits, or sequences of multiple digits, with '0', replacing capital letters with lowercase letters, removing the newline symbol, punctuation, trailing and leading spaces, and removing repeated space symbols. Additionally, stop words were removed to limit the size of the training dataset for the word embedding model described in Section 3. Forward and backward slashes, as well as hyphens, were replaced with spaces, instead of being removed, to prevent the training dictionary from exploding. These punctuation marks are typically preceded and proceeded immediately by characters, therefore, simply removing them would introduce a large number of words to the vocabulary. For example, the composite word 'farm-related' would become 'farmrelated', whereas the separate words 'farm' and 'related' would already exist in the vocabulary. Finally, sentence strings were separated into words at space symbols.

After the data processing steps mentioned above, the unique words in the training dataset were mapped to integers for ease of handling, and stored in a dictionary. The sequence `<UNK>` was added as a key in the dictionary and represents words not seen in training. During inference, test-set words not seen are replaced with the integer corresponding to the `<UNK>` sequence. The number of unique words, including `<UNK>`, in the training dictionary after pre-processing is 68,457, therefore data processing significantly reduced the vocabulary size. This is exceptionally beneficial given that the study was conducted with limited compute resources.

3. Word Embedding Model

The word embedding model used in this study is a skip-gram with negative sampling. This model was chosen for its computational efficiency over traditional skip-grams or the CBOW model. We sample five negative instances per centre word and use a window size of five. In other words, each centre word corresponds to four context words around it – two on either side. Increasing the number of negative words per centre word would increase contrastive power and enable the model to learn more discriminative embeddings, however, it increases the size of the dataset substantially. Similarly, a wider window would improve model performance by capturing longer-range relationships between words, enriching the embedding space, but this improvement comes at the cost of increasing the dataset size. Therefore, given computational restrictions, we had to limit the number of negative instances per centre word, and the context window, to five. In further analyses, the effect of these hyperparameters can be explored.

The only hyperparameter we considered was whether the embeddings should be normalised before the dot product between the pair of embeddings is taken. Introducing normalisation resulted in poor performance across all of the evaluation methods described in Section 4, therefore the only model considered for the rest of the report is the variant without normalisation.

The skip-gram model with negative sampling was trained for 100 epochs, which was the maximum number of allowed epochs. During training, the early stopping mechanism was not triggered which indicates that the model did not exhibit significant overfitting, and may benefit from further training. The final training and validation binary cross-entropy loss values were 0.753 and 2.694 respectively.

4. Evaluation Methods

In this section we will outline one qualitative and four quantitative evaluation criteria for a trained skip-gram model with negative sampling.

4.1. Visualisation of Dimension-Reduced Embeddings

The first evaluation criterion outlined is qualitative in nature and involves finding the embeddings of a set of key words, performing dimensionality reduction on the embeddings, and then visualising the projected representations. In this study, we found embeddings for words commonly occurring in the corpus across the four news categories, and we used T-distributed Stochastic Neighbor Embedding (t-SNE) to reduce the embeddings to two dimensions for visualisation. These embeddings are shown in Figure 1. We expect clusters of words belonging to the same news category to form, and the figure illustrates some loose clustering; however, there is considerable overlap.

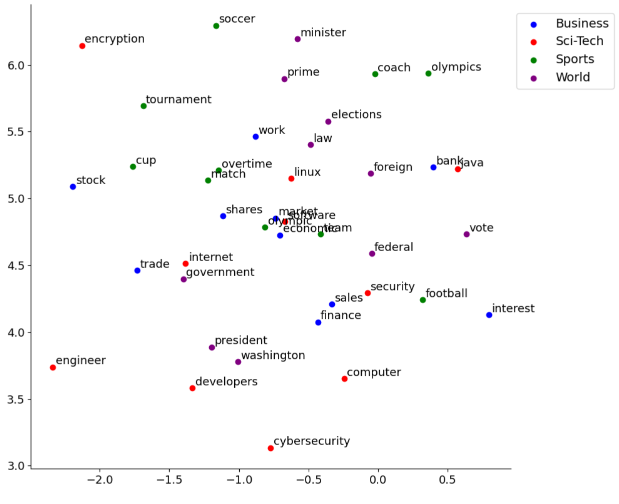

Figure 1. This figure visualises a selection of word embeddings, reduced to two dimensions using t-SNE.

One of the largest drawbacks of this evaluation approach is the loss of information during dimensionality reduction. The original embedding dimension is 512 which was reduced to two dimensions for visualisation, therefore there was a 256-fold reduction in embedding size. As a result of this substantial reduction in dimensionality, some of the finer details of the original word embeddings are inevitably lost. Consequently, words that were initially well-separated in the original latent space may lie close together in the two dimensional t-SNE space, which makes the evaluation of the embeddings close to impossible. This drawback is unavoidable seeing that higher-dimension visualisations are difficult to interpret.

Additionally, attempting to project the entire vocabulary to two dimensions would result in a cluttered visualisation, so we are required to select a set of words. The selected words may not capture the diversity present in the original dataset which could introduce bias into the interpretation of the results.

4.2. Sample Key Cosine Distances

The next evaluation criteria involves identifying sets of words which are semantically similar and comparing the distance between word embeddings from the same set, to the distances between word embeddings from different sets. In this study, four sets of 15 words are identified corresponding to the four news classes. The intra- and inter-class average cosine similarity values are shown in Figure 2. The intra-class mean cosine similarity exceeds, or equals, the inter-class cosine similarity for all classes except science/ technology. Figure 1 supports this finding, as the projected dimensions of the selected science/ technology words lie close to the edges of the embedding spaces and are not tightly clustered. One of the challenges associated with this method is the choice of words per set since the words in the training vocabulary are not confined to a news category. Take, for example, the words ‘team’ and ‘government’. In the previous evaluation method, we assigned the word ‘team’ to the sports category and assigned ‘government’ to the category of world news. However, in the training data the two words are both used frequently alongside the words ‘national’ and ‘US’, therefore the training regime will inadvertently pull the embeddings for ‘team’ and ‘government’ together. The choice of words belonging to each news category greatly impacts the perceived model performance, and as illustrated, selecting representative words is non-trivial.

4.3. Semantic Textual Similarity Benchmarking

One commonly used approach to evaluate word embeddings involves comparing the cosine similarity of pairs of word embeddings to human-annotated perceptions of similarity, which is a task known as semantic textual similarity. Spearman’s rank correlation is used to determine the extent to which the model’s estimation of semantic similarity aligns with human perceptions.

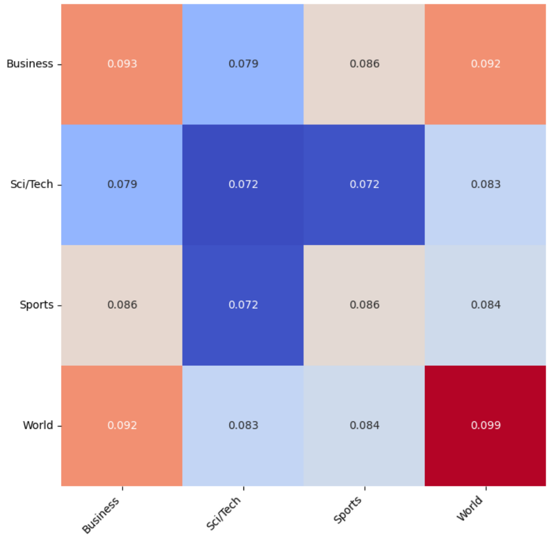

Figure 2. This heatmap illustrates the average cosine similarity between word embeddings related to the same, and to different, news categories.

In this study, we used the WordSim353 dataset as the benchmark for semantic similarity. The dataset contains 353 word pairs with similarity scores between 0 and 10. After filtering out the word pairs where at least one of the words did not exist in our training vocabulary, we were left with 322 word pairs. Our trained model achieved a Spearman's rank correlation coefficient of -0.0434, with a p-value of 0.438 >> 0.05. The correlation coefficient and the large p value indicates that there is no strong relationship between the model's predictions and the human scores, and any correlation may be due to random chance.

The choice of semantic similarity dataset poses a challenge in cases where the training corpus is domain specific, such as the news headlines dataset used in this study. Many of the WordSim353 word pairs would appear infrequently in the context of news headlines, and to our knowledge, there are no publicly available datasets that suit the context of this training corpus. Moreover, the choice of evaluation dataset is complicated by the fact that different evaluation datasets may provide very different similarity scores to word pairs, since the definition of word similarity differs from person to person. For example, the Simlex999 dataset assigns a low rating of 1.96 to the pair *'clothes - closet'*, whereas WordSim353 assigns a high score of 8.0 (Hill et al., 2014).

4.4. Downstream Performance on a Classification Task

The last evaluation criterion considered involves using the trained embeddings in a downstream classification task. Sentence embeddings are derived by averaging the trained word embeddings within a sentence, and subsequently, these sentence embeddings are classified into the four aforementioned news categories. The sentence classifier was trained for 22 epochs before early stopping was triggered, and the results are summarised in Table 1. The test partition confusion matrix is shown in Table 2. While Table 1 and Table 2 illustrate the impressive performance of the sentence classifier, we note that the model is slightly biased towards predicting the 'Sports' class. Table 2 also illustrates that the model may struggle distinguishing between business headlines and science/ technology headlines. The results from Table 1 illustrate that most of the classifier's performance can be attributed to the learned word embeddings, given the impressive accuracy after a single epoch of training.

Table 1. A table showing classification accuracy (%) results across the data partitions, together with the cross-entropy loss.

Data Partition	Accuracy (%)	Loss
Training (Epoch 1)	75.02	0.99
Training (Epoch 22)	86.00	0.88
Validation (Epoch 1)	78.14	0.96
Validation (Epoch 22)	81.83	0.92
Test	81.70	

Table 2. A table showing the confusion matrix between the classes applied to the test dataset.

Predicted	Sports	World	Sci/Tech	Business
True				
Sports	1798	25	39	38
World	195	1482	85	138
Sci/Tech	124	88	1428	260
Business	104	87	208	1501

This evaluation method is a better indication of word embedding performance, within the domain of the training corpus, compared to the semantic textual similarity task (STS) described in Section 4.3. This is because the evaluation criterion is aligned with the domain of the training corpus, whereas the STS dataset is generalised. However, if the generalisability of embeddings is of importance, the STS task should be considered in the evaluation of embeddings.

The advantage of using embeddings in a downstream classifier compared to the evaluation methods in Sections 4.1 and 4.2 is that the entire vocabulary is taken into account, and we do not need to hand-pick a selection of words for evaluation, therefore preventing human-induced bias. One disadvantage of this approach is that it requires a supervised dataset, which may be expensive to obtain.

5. Conclusion

The choice of evaluation criteria for a word-embedding model is not trivial. Methods such as those described in Section 4.1 and 4.2 involve evaluating only a portion of the learned embeddings which may not be representative of all words from the training vocabulary. These methods may be used to detect glaring errors, but should not be considered as more than a preliminary assessment of the model's performance. Similarly, the evaluation of embeddings against a semantic textual similarity dataset only evaluates a small portion of the learned embeddings which may skew the perception of model performance. However, assuming a sufficiently large STS dataset is used, this task may provide a proxy for the generalisability of the learned word embeddings. If generalisability is less important than the performance of embeddings in a domain-specific task, and class labels are available, then a method such as the one described in Section 4.4 will be suitable. Using the embeddings in a downstream classifier ensures that all learned embeddings contribute to the performance score unlike the previously suggested approaches.

Ultimately, the evaluation criteria should depend on the available dataset and the intended usage of the embeddings. However, we recommend semantic textual similarity and downstream tasks as sensible evaluation approaches.

References

Hill, F., Reichart, R., and Korhonen, A. SimLex-999, 2014. https://fh295.github.io/simlex.html. Accessed 24 July 2023.

Wang, B., Wang, A., Chen, F., Wang, Y., and Kuo, C.-C. J. Evaluating word embedding models: Methods and experimental results. *APSIPA transactions on signal and information processing*, 8:e19, 2019.