In a Vector Space of La Mancha, whose Position I do Wish to Recall: A Comprehensive Evaluation of Word Embeddings

María García-Abadillo Velasco
University of the Basque Country
mgarciaabadill001@ikasle.ehu.eus

Abstract

This paper examines the process of collecting and training static word embeddings in a specific domain (the municipality of La Solana) with a focus on their evaluation. Three sets of embeddings (domain-specific, pre-trained fastText, and fine-tuned embeddings) are evaluated both intrinsically and extrinsically. The overall results indicate the superiority of word embeddings with in-domain information when used as features in a domain-specific text classification task. However, they compromise generalization in tasks that examine semantic properties.

1 Introduction

Static word embeddings, also referred to as distributed word representations, were introduced over a decade ago (Mikolov et al., 2013a) as a more advanced architecture for learning word representations compared to traditional count-based methods like Latent Semantic Analysis or Positive Pointwise Mutual Information. Furthermore, they aimed to minimize computational complexity that was present in architectures like Feedforward Neural Net Language models or Recurrent Neural Net Language models.

Since then, new ideas have been proposed to enhance the initial Word2Vec structure. For instance, GloVe combines count-based and prediction methods, while fasttext represents words as bags of character n-grams. More recently, transformers (BERT) have been introduced, which use contextual dependent vectors to capture polysemy and syntactic relationships more accurately than traditional word embeddings.

In this study, we explore static word embeddings by training them on domain-specific data to gain insight into the underlying principles of traditional word embedding designs. Future work may include replicating this project using contextual word embeddings.

2 Sets of Word Embeddings

This paper will evaluate three sets of embeddings: in-domain embeddings trained by us, pretrained fasttext embeddings, and a finetuned version of both.

- **lasolana embeddings**: The corpus was trained using skipgram with 150 dimensions, a window size of 5, and 5 negatives. For a detailed explanation of how the data was collected, refer to section 2.1 and to section 3.1 for details on training.
- **fasttext embeddings**: Trained on Common Crawl and Wikipedia using CBOW, character n-grams of length 5, a window of size 5 and 10 negatives. Their dimensions were reduced from 300 to 150 for our experiments.
- **lasolana+fasttext embeddings**: A fine-tuned set of embeddings obtained with the `fasttext` library in Python.

2.1 In-domain data collection and cleaning

lasolana embeddings, which we also refer as in-domain embeddings, were trained on four main sources related to the town of La Solana:

- **Gaceta de La Solana**: bi-monthly printed newspaper with a tradition spanning over 50 years. Certain editions have surpassed 100 pages, covering a wide range of topics pertaining to the political, societal, and economic aspects of La Villa de La Solana, located in Ciudad Real. 205 PDF files from 1985 until 2023 were obtained from Biblioteca Virtual de Castilla-La Mancha and Ayuntamiento de La Solana converted into TXT files using the open-source command-line utility `pdftotxt`, which was the utility that better performed on this files. However,

a significant portion of the newspaper's structure was lost and required preprocessing and cleaning. Certain issues, such as word concatenation or separation, could not be resolved automatically, resulting in some noise in this subcorpus.

- **La Solana News Dataset**: The dataset comprises 1070 news articles in Spanish extracted from the 'Noticias' section of the La Solana City Council official website. The data has been manually annotated into seven classes. A supervised classification will be performed on this dataset for extrinsic evaluation. More information about the dataset is available in this GitHub repository.
- **Julián Simón's Blogspot**: Julián Simón held the position of town mayor in La Solana from 1983 to 1987 and he was also a teacher. Starting from 2012 until 2019, he authored a digital blog called 'Joaquin Costa. La Solana. Legado Bustillo' with the objective of promoting the history and cultural heritage of La Solana. The entries of this blog are characterized by their linguistic precision and use of specialized in-domain terminology related to the town's traditions and culture. This makes the data high-quality for our training purposes.
- **El Legado Bustillo Book**: The book about Legado Bustillo dates back to 1935 and was written by Jose María García Gallego, a doctor who wrote about a controversial lawsuit that occurred in La Solana at the beginning of the last century. The lawsuit brought the acclaimed politician and lawyer Joaquín Costa to the town. Not much is known about the author beyond this.

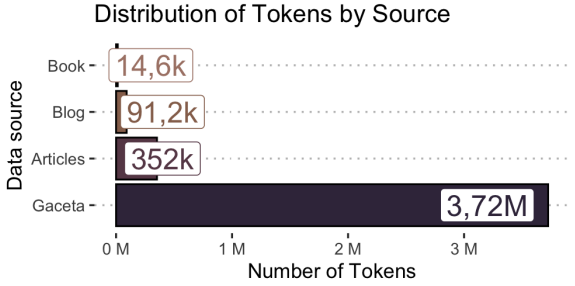

Figure 1: Distribution of tokens according to their source.

There is a significant difference in the number of tokens added by each source to the corpus. Statistics about this can be found in the GitHub repository for this project.

Finally, the corpus composed of these four subcorpora was combined and split into sentences. This input will be used to train word embeddings using the `fasttext` library.

3 Embedding Training

In a Word2Vec skipgram model (Mikolov et al., 2013a), word embeddings are trained by teaching them to predict contexts. The objective is to ensure that each embedding captures the context of the word it describes, aligning with the distributional hypothesis that words that appear in similar context have similar meanings. Throughout the training process, the parameters to be learned are the embeddings themselves. These parameters undergo iterative updates as a n-size window slides over the text. The objective or loss function first identifies potential contexts within the window and the probabilities are calculated using a softmax function. Then, at each step, the target vector (the embeddings to be retained) and the context vector (which will not be needed once training is complete) are optimized using gradient descent. To expedite training, negative sampling (Mikolov et al., 2013b) is applied to the context vector, updating only a set number of k vectors. The optimization process increases the similarity between words that appear in similar contexts and decreases it when they do not.

3.1 Hyperparameter tunning

The `fasttext` library enables training embeddings with just one line of code. As a result, we performed an initial 'in-domain' analogy test on the corpus presented in section 2.1 using different hyperparameter combinations to identify the best set for our final embeddings. Accuracy was not recorded in this preliminary evaluation. Unless otherwise specified, we followed the findings in Lai et al.

- Skipgram vs. CBOW: No test was conducted to compare both models due to time constraints, since 'in case of smaller corpus, a simpler model, such as Skip-gram, can achieve better results'.
- Subwords: The embeddings performed better in the preliminary analogy test without subwords and therefore 'minn' and 'maxn' hyperparameters were set to 0 as in default modus.

- Dimensions: According to Lai et al., 'for tasks that analyze the semantic properties of word embeddings, larger dimensions can provide better results. (...) [A]s feature or for initialization, a dimensionality of 50 is sufficient'. Despite this, the dimensions were set to 150 because they performed well in the analogy test and were not expected to negatively impact the classification test.

- Epoch: It was observed that using a default value of 5 epochs improved results compared to using a smaller number of epochs, but stagnated with a larger number.

- ws: Refers to the size of the context window. It was set to 5 as in Mikolov et al. 2013a.

- neg: Refers t0 negative sampling. Mikolov et al. 2013b show that setting this hyperparameter to 5 'achieves a respectable accuracy' and did further experiments with neg=15 that obtained 'a considerably better performance'. We kept the default setting at 5 because there were no substantial differences in results, and the initial approach was less time-consuming.

4 Evaluation

The training of word embeddings is unsupervised because it does not rely on labeled data or human annotations. Therefore, word vectors can be evaluated intrinsically by measuring how well they capture semantic and syntactic relationships between words, or extrinsically on a downstream NLP task, which involves labeled datasets and supervised learning techniques. We will carry out both.

4.1 Intrinsic evaluation

The intrinsic evaluation aims to assess the embedding's semantic properties and it was divided in two parts:

4.1.1 Analogy test

Inspired by the well-known analogy test first published by Mikolov et al. 2013a entitled Google analogy test, a much more modest test was created for our purposes, consisting of 20 question pairs. Ten of them are domain-specific pairs and the rest are general pairs.

4.1.2 toefl test

Following Lai et al., 2015, we created a set of questions that simulate the TOEFL test. In their experiment, the set contains 80 multiple-choice synonym questions with 4 candidates each. In our case, the set contains 21 multiple-choice questions with 4 candidates each, of which 10 are in-domain and 11 are general.

word1	word2	word3	word4
fútbol	ff_la_solana	baloncesto	cb_la_solana
tener	tenido	vivir	vivido

Table 1: Selected example of analogy pairs. The whole set can be found on Github.

word1	word2	word3	word4
capacho	estera	espuerta	alquiler
vehículo	coche	automóvil	toros

Table 2: Selected example of toefl words. The whole set can be found on Github.

4.2 Extrinsic evaluation

Extrinsic evaluation shifts the focus towards evaluating embeddings based on their performance in downstream NLP tasks. In our case, performing text classification on La Solana News Dataset, which has seven predictors.

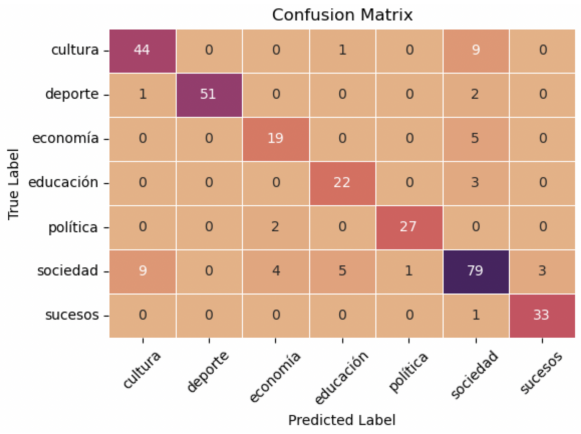

Figure 2: Confusion matrix of the Random Forest classifier when initialized with finetuned embeddings as features.

We will use a Random Forest classifier, as it was the classifier implemented in our last project for the Machine Learning II course on the same corpus (this will be published in GitHub soon). Our goal is to compare the results of the three set of embeddings we intrinsically evaluated before, with that of the previous MLII course that did not use embeddings.

5 Results

The analysis of the results does not reveal a definitive superior candidate among the evaluated sets of word embeddings. However, it provides interesting insights into the potential applications of each one.

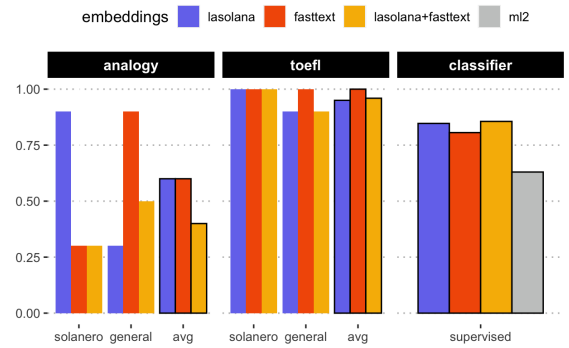

Figure 3: Results of intrinsic evaluation (with visualization of in-domain and general accuracy) and extrinsic evaluation

On the one hand, the first graphic illustrates the results of the intrinsic evaluation task for analogies. It shows that both the in-domain embeddings (lasolana) and pretrained embeddings (fasttext) have the same accuracy (average column with black borders), while the finetuned set performs worse. However, upon examining the 'solanero' and 'general' columns, it becomes apparent that their performance is exactly opposite: in-domain embeddings perform well in the in-domain pair of analogies, while the pretrained embeddings perform poorly. On the contrary, when testing general pairs of analogies, pretrained embeddings perform well, while in-domain embeddings perform poorly. This leads to the same accuracy when taking into account the 20 set of questions, but there is an interesting underlying difference. On the other hand, the second graphic displays the performance of the embedding sets in the second intrinsic task. It is evident that all sets of embeddings were able to master this task, especially the pretrained ones.

Finally, the text classification task shows that approaches using word embeddings outperform the approach that uses a document-term matrix (grey column with an accuracy of 0.63). However, the difference among word embedding sets is relatively small. The best set is the finetuned one with an accuracy of 0.85, while the in-domain embeddings are just one point below. The pretrained set is the worst with an accuracy of 0.80. This means an improvement of 35% in accuracy when using word embeddings as features instead of a document-term matrix. Moreover, as Figure 2 indicates, the class 'sociedad' appears to be the most difficult topic to predict. This aligns with our expectations since it was used as an umbrella topic during annotation, as explained here. Therefore, it could be argued that the inaccuracies are not directly caused by the use of embeddings as features, but rather by the challenging nature of the dataset.

6 Conclusions

This paper documents the training and evaluation process for three sets of embeddings: in-domain embeddings trained on collected data, Spanish pretrained fasttext embeddings, and a fine-tuned version of both. The results indicate that training domain-specific embeddings on only 4 million words was successful, but do not clearly show a superior set of embeddings, as their application depends on the context. The evaluation revealed that finetuned embeddings are effective for text classification when the data being classified is from the same domain. In-domain embeddings would be suitable for dissemination purposes, such as presenting our project in the town of La Solana as a demonstration in an NLP workshop. Finally, FastText embeddings are the optimal choice for general tasks.

7 Discussion

Although we believe to have conducted a comprehensive evaluation of the word embeddings trained on the collected corpus, including both intrinsic and extrinsic tasks, this paper does not present the whole picture, as we do not examine state-of-the-art embedding structures. Further work aims to replicate this study using BERT or ELMo embeddings.

References

Siwei Lai, Kang Liu, Liheng Xu, and Jun Zhao. 2015. How to generate a good word embedding?

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

Tomás Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546.