Topic-dependent Word Embeddings

Authors: *Ziming Dong, Kaushik Manchella, Luke Snyder*

1 Abstract

Dense vector representations of words have been increasingly used in natural language processing (NLP) tasks due to their ability to capture meaningful linguistic semantics at the word level. However, current approaches for learning these representations do not account for the topical context, which can provide useful information and therefore improve representation quality. In this project, we trained word embedding models on topic-dependent data corpora to incorporate topical context. We then evaluated the topic-dependent embeddings by comparing their performance against topic-independent embeddings on several downstream classification tasks.

2 Introduction

The core intuition behind our project is that a word may convey different semantic information across different topics. A simple example is with the word "apple" which conveys different semantic information when the topic of discussion is "fruit" or "computers". With traditional distributed word representations such as Word2Vec [8], GloVe [9], and Fastext [4], "apple" always has a singular vector representation that is applied in all scenarios. This is a wasteful approach in representation when considering that we want high performance on a downstream application. There are many methods that currently exist in disambiguating the sense in which a given word is being used with contextual representations such as ELMo [10] and BERT [5]. However, these methods still do not account for the overall topical context during training and often require heavy computational resources to implement and fine-tune for a given downstream application.

We developed and evaluated a straightforward heuristic method of learning word representations specific to different topics. We aim to answer the primary question of whether using topic-specific word embeddings is a viable method to improve performance on downstream classification tasks. In assessing the performance, we compare our method with two baselines: Word2Vec and BERT. We first built and compared topic-dependent and topic-independent Word2Vec representations to evaluate how much of an improvement topic-specific word representation may provide. We then implemented and compared against BERT-base, which utilizes a highly contextual word representation and has proven to perform well on disambiguation of word senses across a variety of tasks. We implement this baseline as it allows us to compare our approach to the basis of current state-of-the-art NLP systems.

2.1 Novel Aspects

The novel aspects of this project can be summarized from the following perspectives:

1. We provide a novel approach to learn topic-dependent word embedding models by splitting data corpora by topics and training a model on each topic-dependent data subset.
2. We demonstrate the effectiveness of topic-dependent embeddings by comparing their performance with topic-independent embeddings on downstream sentence classification tasks.

3 Problem Definition

The over-arching classification task that we are evaluating with our word representations can be mathematically formulated as follows:

$$\begin{aligned} \mathbf{e} &= W^e(\mathbf{w}) \\ \mathbf{x} &= \phi(\mathbf{e}) \\ P(\mathbf{y}|\mathbf{x}) &= W_1 * \mathbf{x} + b_1 \end{aligned}$$

We denote the input word token vector by $\mathbf{w}$. Vector $\mathbf{w}$ is of length T which is equivalent to the length of the input sentence/document. $\mathbf{e}$ is the embedding representation of our input sentence/-document. Consequently, $\mathbf{x}$ is the set of extracted features which is used for our final classification task to get $P(\mathbf{y}|\mathbf{x})$. The final classification performance is observed for three separate tasks including relevant disaster tweet classification (2 classes), irony detection (2 classes), and hyper-partisan news detection (2 classes). Performance for the tasks mentioned are evaluated with our topic-based word embedding method as well as our baseline word embedding methods which include Word2Vec (Skip-gram) and BERT-base features.

The task at hand is to solely evaluate the different methods' ability to encode words with the right semantic meaning, hence the only part of our equation set that is modified across the different methods is $\mathbf{e} = W^e(\mathbf{w})$, where W^e is the embedding lookup unique to each of the representation methods (and for each topic, as discussed in the following section). This allows use to make fair comparisons between all methods.

4 Technical Approach

Our technical approach consists of two primary components: (1) learning topic-independent embeddings (as a baseline) and (2) learning topic-dependent embeddings.

We first experimented with topic-independent and -dependent embeddings trained on a large Wikipedia corpus [1] (approximately 17GB). However, the topic-dependent performance on downstream tasks was poor, which we expect is because the topics learned on the Wikipedia corpus are not representative of the topics in the downstream dataset. As such, we decided to train and fine-tune the word embedding models on the downstream datasets for a more meaningful evaluation.

In training the traditional (topic-independent) models (for baseline comparison), we first used the standard skip-gram Word2Vec model provided by Gensim [8]. We trained this model from scratch on each downstream dataset. For the next baseline, we used the BERT-base model. Due to the computational resources and time required for training a BERT model from scratch, we fine-tuned the pre-trained transformer architecture parameters [5] for each dataset. The hidden states from the final transformer layer were used as word representations for downstream classification.

In training the topic-dependent models, we followed a similar approach that also incorporates the topical context. Our method is outlined as follows:

1. We first trained a topic model on each downstream dataset; we experimented with the Latent Dirichlet Allocation (LDA) [3] and Non-negative Matrix Factorization (NMF) [7] methods.
2. After training the topic model, we split the dataset into topic subsets. We accomplished this by passing a given sentence/document through the topic model, obtaining its topic distri-

bution, selecting the most prevalent topic from the distribution, and saving the sentence/-document to that topic's corpus. The result is the initial dataset now split by topics.

3. We then trained the word embedding model (from scratch with Word2Vec and fine-tuned with BERT) on *each* topic subset. The result is an embedding model trained for each topic.

To use the topic-dependent models for downstream classification, given an input sentence/document $\mathbf{w}$, we first computed its most prevalent topic t with the learned topic model. We then obtained its topical embedding representation $\mathbf{e} = W^t(\mathbf{w})$ by retrieving the word representation of each word $w_i \in \mathbf{w}$ from the Word2Vec/BERT model trained on topic t.

It is important to note that for Word2Vec, we also experimented with fine-tuned topic-dependent embeddings. We found that for *some* downstream datasets, the topical embeddings did not perform well, possibly because the topic-split subsets contain less data than the entire downstream dataset. To remedy this, we used the trained topic-independent Word2Vec model (which we optimized with validation data) as a starting point to train the topic-dependent Word2Vec models.

5 Evaluation

5.1 Rationale

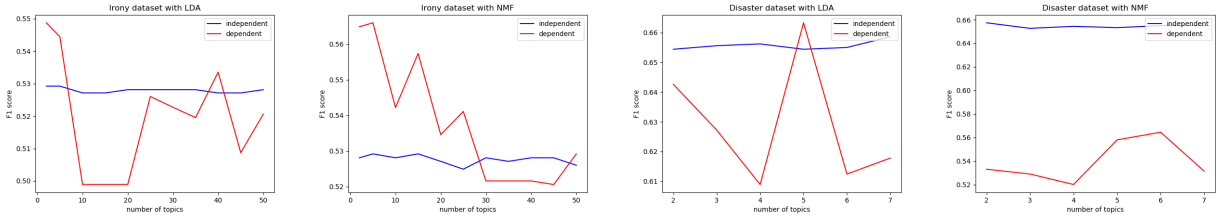
The main research question we wanted to answer is: do topic-dependent embeddings outperform topic-independent embeddings (baseline)? Specifically, we want to investigate how topic-dependent embeddings can provide improved word representations over traditional topic-agnostic embedding approaches. To evaluate, we compare the performance using topic-dependent embeddings and topic-independent embeddings on several downstream classification tasks.

5.2 Experimental Settings

We experimented with four binary classification tasks: (1)-(2) Nepal and Queensland disaster tweets [2], (3) Twitter irony [11], and (4) hyper-partisan news bias [6]. As discussed in the technical approach, we trained a (1) topic-independent model and (2) topic-dependent model on each dataset using both Word2Vec and BERT. For Word2Vec, we trained both topic-dependent embeddings from scratch and with fine-tuning on the topic-independent embeddings. For each dataset, we split it into 60/20/20 training/validation/testing sets and trained a logistic regression (LR) classifier with topic-independent embeddings (baseline) and topic-dependent embeddings. The validation data is used to optimize the LR hyperparameters (e.g., regularization, iterations, etc.). In addition, we averaged the word vectors over the entire input sentence/document.

5.3 Results

Number of topics as a parameter:
Word2vec word embeddings without fine-tuning on irony and nepal disaster datasets:



Classification performance:
TI = Topic-independent; TD = Topic-dependent (using LDA)
TD w/ FT = Topic-dependent with fine-tuning (using LDA)
TD w/o FT = Topic-dependent without fine-tuning (using LDA)

Word2Vec						
Dataset	Accuracy			F1		
	TI	TD w/ FT	TD w/o FT	TI	TD w/ FT	TD w/o FT
Nepal disaster	64.91%	60.53%	67.57%	70.54%	67.76%	73.78%
Queensland disaster	91.71%	92.07%	80.10%	91.31%	91.81%	81.91%
Irony classification	50.98%	52.28%	53.46%	47.92%	51.20%	63.76%
Bias classification	85.67%	86.83%	84.33%	91.67%	92.23%	91.03%

BERT				
Dataset	Accuracy		F1	
	TI	TD	TI	TD
Nepal disaster	74.96%	75.77%	74.26%	75.09%
Queensland disaster	94.97%	95.62%	94.31%	90.20%
Irony classification	61.20%	59.74%	59.78%	54.58%
Bias classification	95.70%	94.47%	91.26%	87.37%

6 Summary

Overall, we provided a novel approach to word representation by incorporating topical context into the learning process. We evaluated the topical embeddings on several classification tasks, demonstrating increased performance over traditional topic-independent embeddings. We learned that topical context can improve word representations. BERT outperformed all methods; however, topic-based fine-tuning on BERT yielded negligible improvements. We attribute this to the robustness of BERT's "bidirectional masked language model", which is already adept at capturing different word senses without topic-based fine-tuning. However, BERT requires significant computational resources, and therefore, may be less desirable compared to topic-dependent Word2Vec for production NLP systems. For future work, we would like to explore topic-dependent performance with more advanced classifiers, such as Bi-LSTM, and using input embedding features that are not averaged over the entire sentence/document.

6.1 Difference from proposal

The project turned out well. The only major change we made compared to the proposal is training the embeddings on the downstream datasets as opposed to on a large Wikipedia corpus since, as discussed previously, the topic splits largely depend on the dataset.

7 Team member contribution

Ziming Dong: Preprocessed irony classification dataset; experimented with dataset corpora workflow to see how the number of topics affects performances.
Kaushik Manchella: Preprocessed hyperpartisan dataset; Implemented topic-modeling pipeline for BERT Fine-tuning for each of the tasks. (refer to TopicBased-BERT-FineTuning.ipynb)
Luke Snyder: Preprocessed disaster datasets; implemented topic-dependent workflow with Wikipedia corpus (for initial experimentation); implemented topic-dependent workflow with disaster datasets.

References

[1] https://dumps.wikimedia.org/enwiki/latest/.

[2] F. Alam, S. Joty, and M. Imran. Domain adaptation with adversarial training and graph embeddings. 2018.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[4] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[6] J. Kiesel, M. Mestre, R. Shukla, E. Vincent, P. Adineh, D. Corney, B. Stein, and M. Potthast. SemEval-2019 task 4: Hyperpartisan news detection. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 829–839, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics.

[7] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.

[8] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[9] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[10] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[11] C. Van Hee, E. Lefever, and V. Hoste. SemEval-2018 task 3: Irony detection in English tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 39–50, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.