Word Embeddings for Constructive Comments Classification

Diego Uribe
TecNM Instituto Tecnológico de La Laguna
División de Estudios de Posgrado e Investigación
Blvd. Revolución y Cuauhtémoc, Torreón, Coah., MX
E-mail: diego@itlalaguna.edu.mx

Abstract. Word embeddings are so significant today that it is common to see their application in multiple natural language tasks. Indeed, word embeddings as the first layer of a deep learning model are widely adopted and they can also be found in multiple natural language tasks such as classification of texts and named entity recognition. The focus of this paper is the identification of constructive online comments through the use of dense vector semantics such as word embeddings. We specifically explore two approaches: learning distributed word representations and the use of a pre-trained word embedding model. We evaluate these word embedding methods on a recently created constructive comments corpus comprised of 12,000 annotated news comments, intended to improve the quality of online discussions. The obtained results show how the performance of complicated architectures like recurrent and convolutional neural networks can be matched by a language model based on learning embeddings.

1. Introduction

We are interested in this paper in the analysis of online comments on the articles published in online newspapers. Since the web makes possible to articulate instant comments and immediate feedback as well, an online discussion forum is encouraged. As the quality of a discussion depends on reasonable comments, that is, comments that exhibit argument quality, coherence and readability, these sort of comments are the target comments to be detected in this work. Below we show examples of a constructive (C) and a non_constructive (NC) comment on an article about the Canadian government censorship that prevents its scientists from talking about sensitive topics such as climate change.

(C) A scientist colleague told me that at a meeting of a committee of one of the Federal govt science funding agencies there was a govt minder present who only commented or ‘intervened’ on decisions when certain topics were raised; climate change and anything to do with the tar sands. If true (its hear-say - not trying to be coy; wasn’t me), this shows the magnitude of the muzzling of scientists that is occuring in Canada. Not content to dictate what govt scientists say, scientists at Canadian universities may...
lose their funding if their research is not deemed ideologically correct. I feel a chill wind, and its name is fascism.

(NC) The sad reality is that most Canadians, as long as they can watch Don Cherry on Sat nite and as long as Tim Hortons keeps their donut prices affordable, they simply don’t care about access to information...

The first comment (C) has been classified as constructive comment since it exhibits constructive characteristics such as: readability: each sentence targets a specific point; evidence: first sentence provides evidence, and dialogue: it contributes to the conversation. On the other hand, the second comment (NC) has been classified as non-constructive comment since it exhibits non-constructive characteristics such as: sarcastic: it hurts someone’s feelings or it exhibits humorous criticism; unsubstantial: it lacks of significant argument, and not relevant: it does not contribute to the conversation.

These examples illustrate how important is the necessity to separate misleading information that does not enrich discussions. Actually, enriched discussions demand the need to identify constructive comments to improve the quality of the information and in this way, as Vinton G. Cerf argues, to learn to become our editors in the use of the information that surround us [1].

To automatically recognize constructive comments entails the use of proper data, that is, the use of a suitable repository of annotated comments. Fortunately, we have nowadays a repository specifically developed for the production of tools to identify constructive comments. This corpus comprised of 12,000 annotated news comments is described in section 3.

2. Related work

The use of word embeddings is very common in multiple natural language tasks. As we previously said, not only the use of word embeddings as the first layer of a deep learning model has widespread adoption, but also the use of word embeddings in multiple natural language tasks such as classification of texts or named entity recognition. Martina Toshevska et al. focus their attention on the use of word embeddings to capture word similarities. In this work, they analyse multiple word embeddings methods by making use of existing benchmark datasets for word pairs similarities [2].

Kolhatkar and Taboada implemented classical (SVM classifiers) and deep (biLSTMs) learning models to recognize constructive comments by making use of two datasets for training: the New York Times Picks as positive examples and the Yahoo News Annotated Comments Corpus as negative examples of constructive online comments. The learning models were evaluated on a crowd-annotated corpus (the preceding corpus of C3) containing 1,121 comments [3]. Another previous work implemented by Kolhatkar and Taboada explored whether an argumentation corpora could be useful to identify constructive comments. A deep learning model (biLSTMs) was defined to recognize constructive comments by making use of two datasets for
training: the Yahoo News Annotated Corpus and the Argument Extraction Corpus that includes annotations for argument quality on sentences [4].

3. Data

The corpus used in our work, known as Constructive Comments Corpus (C3), is a collection of 12,000 online news comments with metadata information about constructiveness and toxicity [5]. The distribution of comments to classes represents an almost balanced dataset: 6,516 constructive comments (54%) and 5,484 non-constructive comments (46%). This class distribution is enough to make use of classification accuracy as performance metric for evaluating our classification models. It is also important to mention that the comments to be submitted to the annotation process were obtained from the SFU Opinion and Comments Corpus (SOCC) which contains a collection of opinion articles and the comments posted by readers in response to the articles [6].

4. Word embeddings models

4.1. Learning Word Embeddings model

The purpose of this model is to obtain the dense vectors that characterise the collection of comments. More specifically, these dense vectors are defined by learning representations of the meaning of the words directly from their distributions in the corpus. In contrast to traditional learning methods that demand more pre-processing of the texts for the definition and extraction of features, learning representations directly from the texts denotes a self-supervised learning method that requires a minimum preprocessing (e.g. a simple one hot encoding process).

4.2. Pre-trained GloVe Word Embedding model

In this case, the model is based on a pre-trained word embedding method, that is, instead of learning the vector representation, these dense vectors have already been defined. In this work, we make use of a pre-trained word embedding method known as GloVe: an unsupervised learning algorithm for obtaining vector representations for words. This method is based on a log-bilinear regression model that combines the advantages of two models: global matrix factorization and local context window methods [7].

5. Experimental Evaluation

5.1. Learning Word Embeddings model

In this case, we learn a word embedding by using the Embedding layer of Keras. In other words, by using this embedding method, that it is also commonly used as the first layer of a deep learning model, we learn a distributed representation corresponding to
Table 1. Learning Word Embeddings model results

| output vector | F1  |
|---------------|-----|
| 50            | 93.28 |
| 100           | 93.69 |

Table 2. Pre-trained GloVe Word Embeddings model results

| output vector | F1  |
|---------------|-----|
| 50            | 93.22 |
| 100           | 93.31 |

the vocabulary of the entire corpus of comments. We implement two experiments with this model by keeping two parameters fixed (vocabulary size and input sequence length equivalent to the longest comment) and testing different values for the representation of a word. One experiment represents each word with an output vector size of 50 whereas the other one makes use of a vector space of 100 dimensions. We also split the 12,000 comments in a proportion of 80% for training and 20% for testing purpose. The obtained results with the Word Embedding classifier are shown in Table 1 where we can see how a higher F1 score is obtained with a larger output vector.

5.2. Pre-trained GloVe Word Embedding model

In this case, we make use of the pre-defined embedding method known as GloVe. In particular, we use the smallest package of embeddings called glove6B.zip (862 Mb) that was trained on a dataset of six billions of tokens with a vocabulary of 400K words. The package contains multiple embedding vector sizes, so embeddings of 50, 100, 200 and 300 dimensions are available (https://nlp.stanford.edu/projects/glove/). Since the Keras Embedding layer can also use the word embeddings of a pre-trained embedding method, we seed this layer with the pre-trained embeddings for the words corresponding to the C3 corpus. We also implement two experiments with this model by keeping two parameters fixed (vocabulary size and input sequence length equivalent to the longest comment) and testing different values for the representation of a word. The obtained results with this word embedding method are shown in Table 2 where we can see how similar F1 scores are obtained in both experiments.

To put in context our experimentation, we first compare the performance between our models, and then, our results are compared with the deep learning models implemented by the creators of C3 [5]. As Tables 1 and 2 show, the obtained results are almost similar. However, learning embeddings provides the best results in the identification of constructive comments. Now, about the experimentation with deep learning models, the authors of C3 implemented three learning techniques: Long Short-Term Memory networks (biLSTMs), Convolutional Neural Networks (CNNs) and
pretrained Bidirectional Encoder Representations from Transformers (BERT). From these deep models, the best result was obtained with BERT: a F1 score of 0.93 is reported. In our case, as we previously mentioned, we learn a word embedding, that is, we learn a distributed representation of the entire corpus. The obtained results are shown in Table 3 and as we can see, we achieve the same performance but with a more efficient learning model. In other words, since the full attention mechanism of BERT is very demanding in terms of memory [8], our work represents not only a plausible but also a more efficient learning model.

6. Conclusions

In this paper, we have analysed word embedding methods to identify constructive online comments. We first explore the use of a a language model based on learning vector space representations of words. Then, we have analysed a language model based on the use of a pre-trained word embedding method such as GloVe. In this way, by learning representations of the meaning of words directly from their distributions in the comments, we have demonstrated how our model was able to match the performance of complicated architectures like recurrent and convolutional neural networks.

[1] Vinton G. Cerf, “Can Liberty Survive the Digital Age?,” Communications Of The ACM, vol. 60, num. 5, pp. 7–7, May 2017.
[2] Martina Toshevska, Frosina Stojanovska and Jovan Kalajdjeski, “The ability of word embeddings to capture word similarities,” International Journal on Natural Language Computing (IJNLC) Vol.9, No.3, pp. 25–42, June 2020.
[3] Varada Kolhatkar and Maite Taboada, “Using New York Times Picks to identify constructive comments,” Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism, Copenhagen, pp. 100–105, 2017.
[4] Varada Kolhatkar and Maite Taboada, “Constructive language in news comments,” Proceedings of the First Workshop on Abusive Language Online, Vancouver, pp. 11–17, 2017.
[5] Varada Kolhatkar, Nithum Thain, Jeffrey Sorensen, Lucas Dixon and Maite Taboada, “Classifying Constructive Comments,” arXiv, April 2020.
[6] Varada Kolhatkar and Hanhan Wu and Luca Cavasso and Emilie Francis and Kavan Shukla and Maite Taboada, “The SFU Opinion and Comments Corpus: A Corpus for the Analysis of Online News Comments,” Corpus Pragmatics, vol. 4, pp. 155–190, 2019.
[7] Jeffrey Pennington, Richard Socher and Christopher D. Manning, “Glove: Global vectors for word representation.” Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.
[8] Manzil Zaheer and Guru Guruganesh and Avinava Dubey, “Big Bird: Transformers for Longer Sequences,” arXiv, July 2020.