Comparing Text Vectorization Techniques for Sentiment Analysis Task

732A92 Text Mining Project Report

Bayu Brahmantio (baybr878)

March 10, 2021

1 Introduction

Text classification is a recurring problem in the natural language processing field. One of its many applications is sentiment analysis which tries to identify the subjective information in a given word or text. The usual task for sentiment analysis is classifying polarization of a text e.g. whether a text has positive or negative sentiment.

Generally, given a collection of N documents paired with its class, $(d_1, c_1), (d_2, c_2), ...,$ (d_N, c_N), we want to find a classifier f that takes d as an input and gives us a correct class $c \in C$. There is a wide range of classifier suitable for this task. However, we first need to represent the documents numerically before we can feed them into classifiers.

There are many ways to represent documents as vectors. The simplest one is to represent a document as a set of unordered words along with their frequency, also known as *bag-of-words*. In this way, all documents can be represented as vectors of same length in which each element represents a frequency of a term in a given document. The term frequency can also be weighted by its *inverse document frequency* (*idf*) which takes into account the frequency of the term appearing in the collection. The product of term frequency and its *idf* results in *tf-idf* value (Jurafsky and Martin [2009]).

Another way is to represent words as dense vectors of dimension much smaller than the size of vocabulary. Word embeddings techniques like *word2vec* (Mikolov et al. [2013]) allows us to learn similarities between words that can be useful in a classification task. It is also possible to learn contextual embedding as done in *BERT* (Devlin et al. [2019]) where each word can have different representation depending on contexts.

In this project, different techniques for representing texts as vectors will be compared: bag-of-words, tf-idf, word2vec, and BERT. In the case of word2vec, the average of word vectors will be used to represent individual document. In BERT's case, the final hidden state of special classifier token ([`CLS`]) can be used as the document's representation. They will be compared in terms of their performance in a sentiment analysis task using different classifier models: multinomial naïve Bayes, logistic regression, and linear support vector machine (SVM).

2 Theory

3 Data

The dataset used is a collection of 50,000 movie reviews along with its sentiment from Internet Movie Database (IMDB) (Maas et al. [2011]). It is split evenly into 25,000 reviews in the training and test set. There is also a balanced number of negative and positive reviews in each set.

Instead of taking into account all kinds of reviews, Maas et al. [2011] only collected highly-polarized reviews, that is, only reviews that are considered negative and positive are included. A negative review has an IMDB score of ≤ 4 while a positive one has a score of ≥ 7. The scores are in the range of $[0, 10]$. Since it is a case of balanced dataset where the classes are split evenly in training and test dataset, the expected accuracy of a random classifier will be around 50%.

Review	Sentiment
Story of a man who has unnatural feelings for ...	negative
Airport '77 starts as a brand new luxury 747 p...	negative
This film lacked something I couldn't put my f...	negative
Sorry everyone,,, I know this is supposed to b...	negative
When I was little my parents took me along to ...	negative
⋮	
Bromwell High is a cartoon comedy. It ran at t...	positive
Homelessness (or Houselessness as George Carli...	positive
Brilliant over-acting by Lesley Ann Warren. Be...	positive
This is easily the most underrated film inn th...	positive
This is not the typical Mel Brooks film. It wa...	positive
⋮	

Table 1: Overview of the dataset. The reviews shown in the table are from the training set with the top five rows as the first five negative reviews and the five rows below that as the first five positive reviews.

4 Method

5 Results

6 Discussion

7 Conclusion

References

Dan Jurafsky and James H. Martin. *Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition.* Pearson Prentice Hall, Upper Saddle River, N.J., 2009. ISBN 9780131873216 0131873210. URL `http://www.amazon.com/Speech-Language-Processing-2nd-Edition/dp/0131873210/ref=pd_bxgy_b_img_y`.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL `https://www.aclweb.org/anthology/N19-1423`.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/P11-1015`.