CS 6170: COMPUTATIONAL TOPOLOGY

FINAL PROJECT REPORT:
SENTIMENT CLASSIFICATION WITH TOPOLOGICAL SIGNATURES

Prince Osei Aboagye

School of Computing
University Of Utah
SPRING 2019

April 30, 2019

1 Introduction

A common approach in Topological Data Analysis (TDA) is to capture the shape or the underlying structure of shapes in data. Using topology in data science is mostly new, though computational topology and computational geometry existed in applied mathematics for many years. Despite TDA being a new and growing sub-field of data analysis it has achieved significant success in areas such as neuroscience, bioinformatics, sensor networks, medical imaging, shape analysis, computer vision, audio processing and speech analysis.
TDA methods have not been widely applied to natural language processing and subsequently text mining. There is no evidence to believe this is due to the weakness of topology in text processing. Of course it is not easy to define meaningful shapes in textual documents. However when text is interpreted as describing a progression of events (as in movies), topological features, namely, homological persistence, when added to representation of text, can significantly improve classification accuracy.
The goal of this project is show that adding topological features derived from text structure improves classification accuracy.

2 Project Objective

The goals of this project:

- Evaluate the utility of topological features derived from text in traditional Natural Language Processing (NLP) task specifically sentiment classification
- Increase the awareness about topology as a possible source of semantic features in text analytics

3 Related Work

"Persistent Homology: An Introduction and a New Text Representation for Natural Language Processing" a paper written by Dr. Xiaojin Zhu was one of the first applications of persistent homology to natural language processing. The introduced a special kind of filtration known as Similarity Filtration with Time Skeleton (SIFTS). (SIFTS) algorithm identifies holes that can be interpreted as semantic "tie-backs" in a text document, providing a new document structure representation.
Also, Michel et al. 2017 showed in their paper "Does the Geometry of Word Embeddings Help Document Classification? A Case Study on Persistent Homology Based Representations" that using only persistence diagrams for text representation does not seem to positively contribute to document clustering and sentiment classification tasks however when they are combined with other traditional NLP text features they significantly improve classification accuracy.
There is a paper published by Pratik Doshi and Wlodek Zadrozny called "Movie Genre Detection Using Topological Data Analysis" which showed the ability of Topological Data Analysis (TDA) to perform text classification. They showed that TDA not only matches the performance of widely used algorithms like Multinomial Naive Bayes, Logistic Regression for binary text classification, but also can also outperform more advanced techniques like neural networks when in multi-label text classification.

4 Data

The dataset used for the experiment is the Movie reviews dataset obtained from the Internet Movie Database (IMDb) for sentiment analysis. This dataset, containing over 50,000 movie reviews, can be obtained from http://ai.stanford.edu/∼amaas/data/sentiment/, courtesy of Stanford University and A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, Andrew Ng, and C. Potts, and this dataset was used in their famous paper, "Learning Word Vectors for Sentiment Analysis."
We used 50,000 movie reviews from this dataset, which contains the reviews and their corresponding sentiment polarity label which is either positive or negative.

```
In [4]: print( dataset.head())
                                              review sentiment
0  One of the other reviewers has mentioned that ...  positive
1  A wonderful little production. <br /><br />The...  positive
2  I thought this was a wonderful way to spend ti...  positive
3  Basically there's a family where a little boy ...  negative
4  Petter Mattei's "Love in the Time of Money" is...  positive
```

Figure 1: Sample Dataset

5 Text Preprocessing

First of all, we load our movie review data into python. The movie review data is read into python in the form of sentence token. This is then followed by text normalization. After sentence tokenization various other techniques including cleaning text,case conversion, expanding contractions, correcting spellings, correcting repeating characters, removing stopwords and other unnecessary terms, stemming, and lemmatization were performed in other to get the textual data into a form that can be easily understood and interpreted for our classification task.

6 Feature Extraction

Feature extraction is a process whereby we extract meaningful features or attributes from raw textual data or any form of data before we feed it into a statistical or ML algorithm. First of all we provide the movie review as input to the GloVe model which is an unsupervised learning algorithm for obtaining vector representations of words as output. Internally, it constructs a vocabulary based on the input text documents and then outputs vector representations for words. Using various techniques like average weighting or tf-idf weighting, we can compute the averaged vector representation of a document using its word vectors.

7 Methods

TDA methods have not been widely applied to natural language processing and subsequently text mining. This is not to say that that the geometry of word embeddings does not help in text classification.

Of course it is not easy to define meaningful shapes in textual documents. Dr. Xiaojin Zhu in his paper "Persistent Homology: An Introduction and a New Text Representation for Natural Language Processing" presents one of the first applications of persistent homology for natural language processing. His "Similarity Filtration with Time Skeleton" (SIFTS) algorithm identifies holes that can be interpreted as semantic "tie-backs" in a text document, providing a new document structure representation. For more technical detail about the SIFTS algorithm you can read his his paper.
In this project I conducted three experiment. The first one used only 1-dimensional persistent diagram as features derived from text thus the movie reviews for classification. The second experiment was the baseline and this used basically traditional NLP methods like tf-idf, averaged word to extract feature from the movie reviews for classification. Finally, the third experiment combined both features derived from the first experiment and the second experiment for classification. The goal of this experiment was to investigate how well a text classification model built only on 1-dimensional persistent diagram will perform and also how the text classification model will perform if we combine features derived using tradition NLP methods with 1-dimensional persistent diagram obtained from text.

7.1 Method 1: Using 1-dim Persistent Diagram

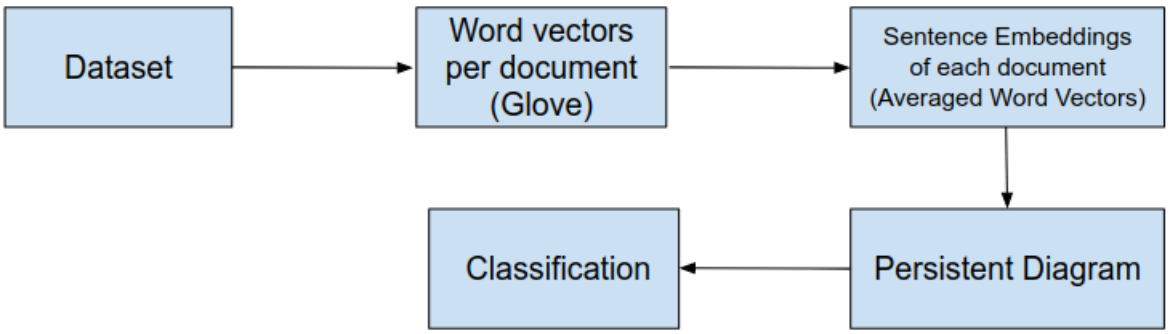

Figure 2: Pipeline

From the pipeline above in order to build a persistence diagram for the movie reviews, we convert each movie review to the set of its word and then feed it into the GloVe model model to learn the vector representation of theses words. Thus the GloVe model Internally, constructs a vocabulary based on the input text documents and then outputs vector representations for words. Since we are using the SIFT filtration we convert each review into sentences. Our goal is to get each review into a vector of n x k where n is the number of sentence in the movie review and k is the embedding size since we are using Glove to get the vector representation of words k will be equal to 300. Now the Glove model on output vector representation of words so the question is how to we use these vectors to get the vector representation of sentences in a movie review. The answer to this quetion is to use the Averaged Word Vectors approach. In this technique, we will use an average weighted word vectorization scheme, where for each sentence we will extract all the tokens of the sentence, and for each token in the sentence we will obtain the word vector using GloVe. We will sum up all the word vectors in the sentence and divide the result by the total number of words matched in the vocabulary to get a final resulting averaged word vector representation for the text document. After obtaining the persistent diagrams using Ripers. These barcodes are then fed into our SVM Classifier for classification.

7.2 Method 2: Baseline

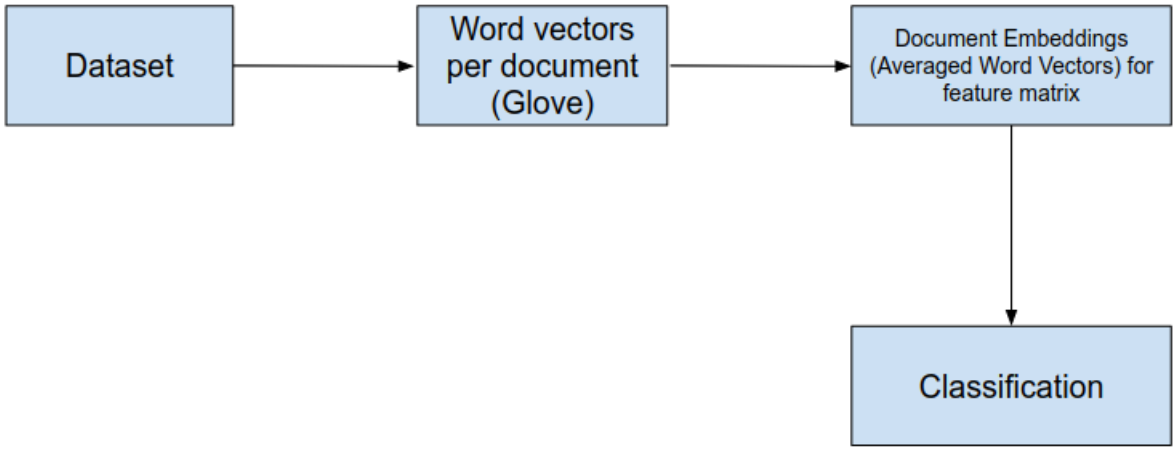

Figure 3: Pipeline

For the Baseline pipeline we use the Averaged Word Vectors technique. However in this case we do not convert each review into sentences but rather we convert the entire review in just a single vector. In this technique, we will use an average weighted word vectorization scheme, where for each text document (movie review) we will extract all the tokens of the movie review, and for each token in the document we will capture the subsequent word vector if present in the vocabulary. We will sum up all the word vectors and divide the result by the total number of words matched in the vocabulary to get a final resulting averaged word vector representation for the text document. The features extracted using this technique are then fed into our SVM Classifier for classification.

7.3 Method 3: Combining both 1-dim Persistent Diagram features and the Baseline features

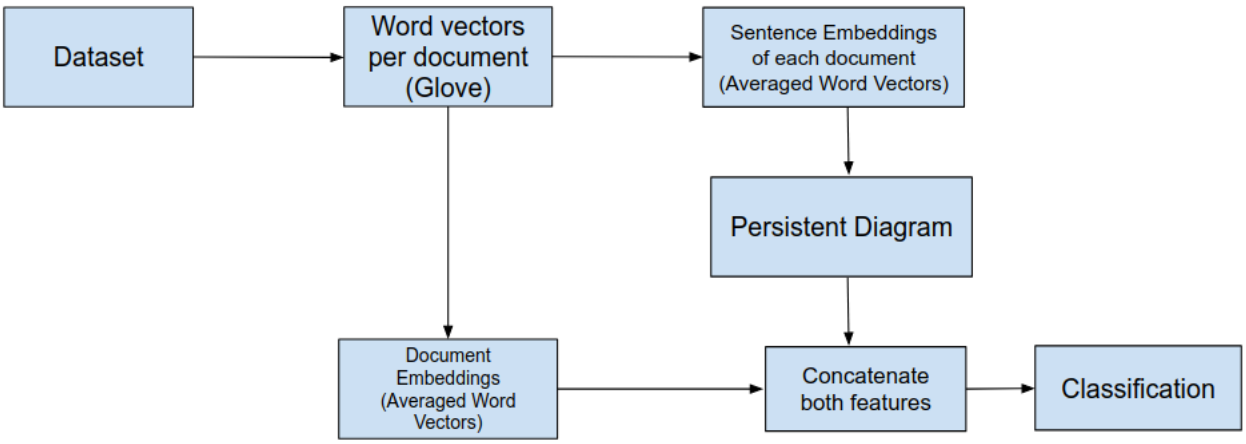

Figure 4: Pipeline

For method 3 we combine both features obtained from method 1 and method 2 and the we feed it into our SVM Classifier for classification.

8 Results: Accuracy

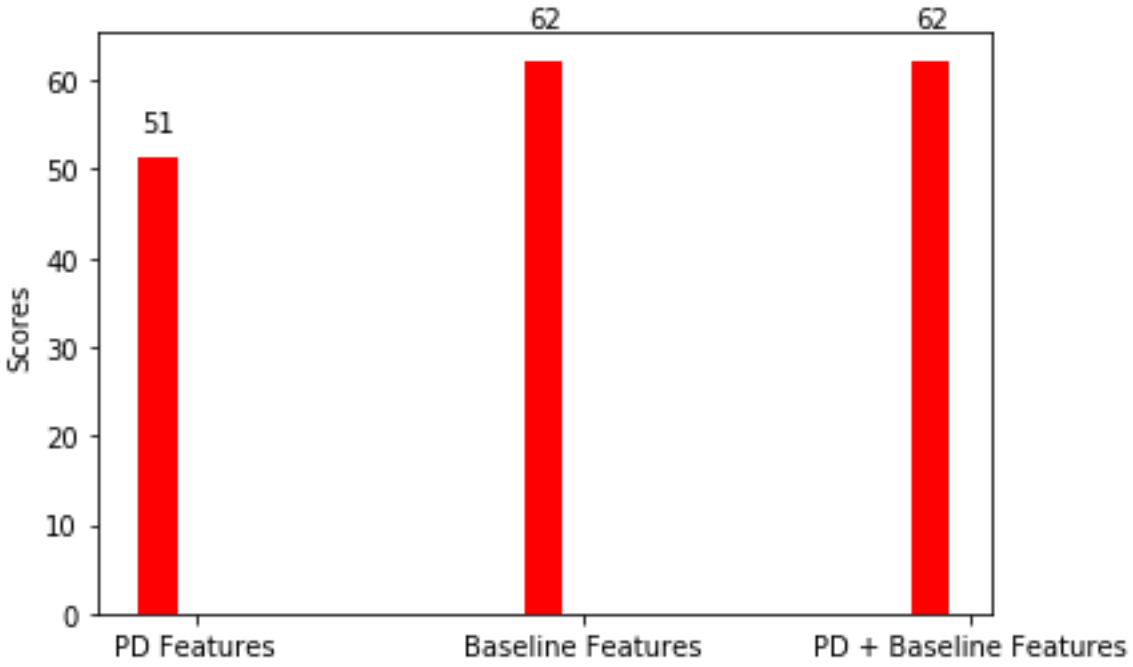

Figure 5: Accuracy on IMDb dataset

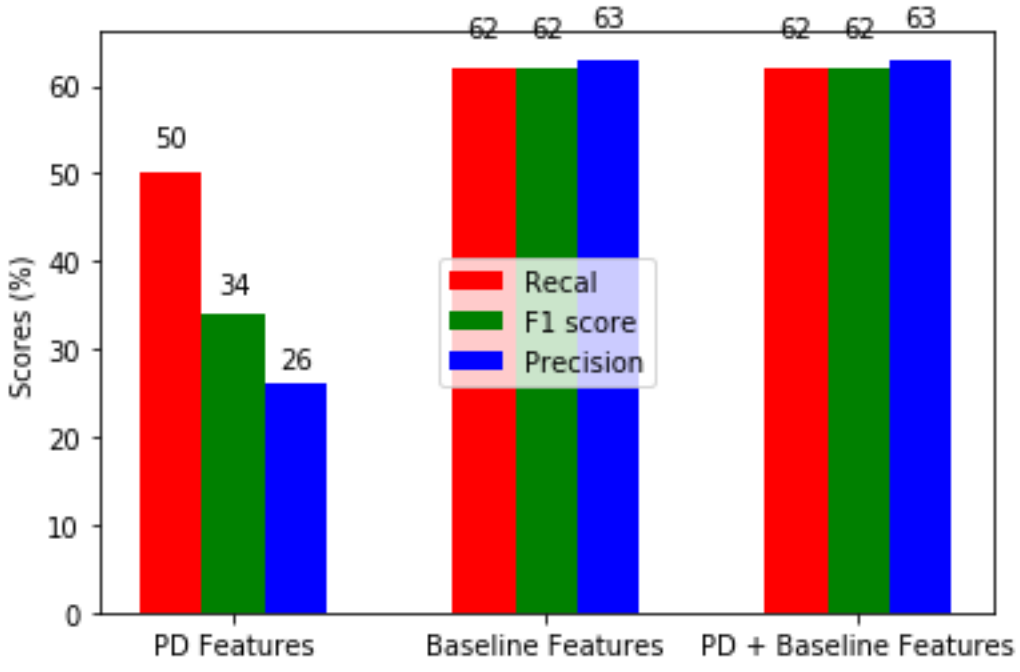

Figure 6: Precision, Recall, F1 score on IMDb dataset

9 Conclusion

Based on our experiments, using persistence diagrams for text representation does not seem to positively contribute to sentiment classification tasks. However Theoretically, algebraic topology has the ability to

capture structural context, and this could potentially benefit syntax based NLP tasks such as parsing. We plan to investigate this connection in the future.

References

[1] Sudipta Kar, Suraj Maharjan, A. Pastor Lopez-Monroy and Thamar Solorio. (2018). *MPST: A Corpus of Movie Plot Synopses with Tags* . CoRR. https://arxiv.org/abs/1801.04813

[2] Doshi P., Zadrozny W. (2018) Movie Genre Detection Using Topological Data Analysis. In: Dutoit T., MartÃn-Vide C., Pironkov G. (eds) Statistical Language and Speech Processing. SLSP 2018. Lecture Notes in Computer Science, vol 11171. Springer, Cham

[3] Gholizadeh, S.; Seyeditabari, A.; Zadrozny, W. Topological Signature of 19th Century Novelists: Persistent Homology in Text Mining. Big Data Cogn. Comput. 2018, 2, 33.

[4] Xiaojin Zhu. 2013. Persistent homology: an introduction and a new text representation for natural language processing. In Proceedings of the Twenty-Third international joint conference on Artificial Intelligence (IJCAI '13), Francesca Rossi (Ed.). AAAI Press 1953-1959.

[5] Qua Hoang. 2018. Predicting Movie Genres Based on Plot Summaries. CoRR. http://arxiv.org/abs/1801.04813