Sentiment Analysis + Dimensionality Reduction On Rotten Tomatoes Dataset

Arshdeep Singh

2019csm1001

Indian Institute Of Technology, Ropar

I Introduction

Natural Language Processing is an important domain of the machine learning applications. Our project tries to provide a comparative analysis of different supervised machine learning algorithms on a popular movie review database, Rotten Tomatoes [1]. The project is divided into 3 modules. The first module performs Exploratory Data Analysis on the dataset. The second module applies different ML models on the dataset for comparative analysis. In the third module, dimensionality reduction is carried out to test out whether we are able to achieve that amount of accuracy with the reduced feature set.

There are basically two popular approaches while representing the text document : Bag of Words (BoW) and Word2Vec. Both the approaches uses numeric methodology for text representation. The simplest is the the bag-of-words approach, where each unique word in a text will be represented by one number. Word2Vec approach implements vectorized projection of word embeddings. For our project, we have used Bag of Words (BoW) approach for representing our review documents.

II Related Work

Scholars have implemented various supervised approaches in NLP. The author in [1] have implemented SVM Classifiers on the multi-class Amazon Review dataset. The implementation has shown interesting comparison of SVM classifiers. Similarly, the authors in [2] applied sentiment analysis on Tweet dataset and reviewed that Multinomial Naive Bayes was able to outperform SVM classifier. Similar classification analysis was done by authors in [3]. The authors in [4] has included Unigrams and Bigrams approache while building the document-term matrix and then the improved feature dataset was fed into the different ML models. The authors in [5] have used unsupervised Paragraph Vector model for text classification.

The document-term matrix in the Bag of Words approach may contain redundant features. That is why scholars have proposed dimensionality reduction on the feature matrix. The authors in [6] have performed PCA (principal component analysis) reduction on popular UCI datasets and have noted a significance increase in the performance of the classifiers after performing feature extraction through PCA. The authors in [7] have performed different dimensionality reduction through LDA (Linear Discrminant Analysis) on the multi-class dataset and have noted a decrease in classification error rate. The reduced feature set also performed well on Word2Vec text model [8]. The authors in [9] have reviewed the performance of different ML classifiers through various dimensionality reduction mechanisms.

III Exploratory Data Analysis

The Rotten Tomatoes movie review dataset is a binary dataset. The reviews are either classified as **Fresh** or **Rotten**. The dataset is available at www.kaggle.com. The dataset has a total 480K samples. To overcome computational constraints, we have performed EDA and experimentations on a reduced dataset. We sampled 100K samples from the dataset. For multiple executions of our experiment, *random_state* was fixed.

The sample 5 data samples from the dataset are:

	freshness	review
227947	rotten	Gone Girl may begin a smart and promising mys...
130796	rotten	One of the most unnecessary sequels of all ti...
308424	rotten	Despite a superb cast, artful set design and ...
389388	fresh	Ssuperbly acted and, for much of its intricat...
399901	fresh	Wonder Woman is glorious.

Figure 1: Sample reviews

[1] https://www.kaggle.com/nicolasgervais/rotten-tomatoes-480000-labeled-critic-reviews

The count of **Fresh**, **Rotten** reviews along with unique labels present in them are tabulated below:

Label	No. of reviews	Unique Words
Fresh	49987	47451
Rotten	50013	45094

Table 1: Total reviews : 100000, Total Unique Labels: 92538

Next, we explored the distribution of the review length. The length of each review was calculated and histogram plot of the same was plotted. The resultant histogram is:

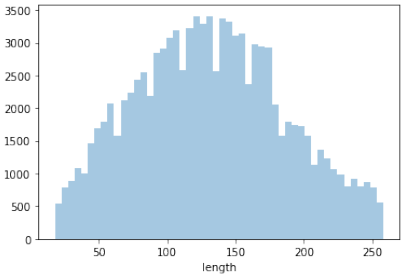

Figure 2: Distribution of review lengths

It is noted that average review length is concentrated around 125-150. The above figure shows the distribution comprising of both **Fresh** and **Rotten** movie labels. Next we analysed the length distribution according to individual labels. The box plot is plotted for that purpose, as it will help us comparing the length distribution of **Fresh** and **Rotten** reviews. It is noted that the length distribution

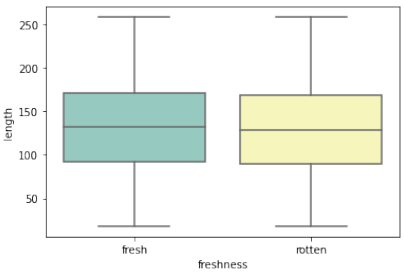

Figure 3: Sample reviews

of **Fresh** and **Rotten** reviews are nearly same. No particular information can be deduced further from the above plot. Finally, various review metrics are shown in the table below.

mean review length	131.47
minimum review length	18.0
maximum review length	258.0
standard deviation	54.66

IV Methodology

This project will expand on the approaches of researches in the past in sentiment analysis. **The project aims to compare the performances of different ML algorithms with and without dimensionality reduction**. We have used **Linear Discriminant Analysis** for dimensional reduction. We have implemented 5 different ML models : Naive Bayes, Logistic Regression, Random Forest, Support Vector Machines and Artificial Neural Network for analysis. Further LDA is applied to the document-term matrix and five models are executed to gain insights about the performance on the **Test Accuracy**.

Data Preparation : Date cleaning and preparation is done through NLTK library [2]. This involves removing punctuation, stopwords and alphanumeric words. The corpora is then converted to the document-term matrix through the use of Sklearn's [3] **TfidfVectorizer** method. This method is a combination of **CountVectorizer** and **TfidfTransformer**. TF-IDF stands for term frequency-inverse document frequency, and the tf-idf weight is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus.

The dataset was split into **Train Set** and **Test Set**. While transforming the corpus to the tfidf matrix, hyperparameters of TfidfVectorizer were tuned. The *min_df* (min. frequency of a word) was set to 10, and *max_features* were set to 3000. That means top 3000 word-features were chosen in the final tfidf matrix. Moreover, Unigram model approach was followed. The dimensions of the train set and test set were respectively : $\mathbf{N \times 3000}$ and $\mathbf{M \times 3000}$.

The performances of the algorithms are obtained in the form of **Classification Report** and **Confusion Matrix**.

V Results : Pre-LDA

The five ML models were applied to the original 3000 feature document-term matrix and classification reports and confusion matrices were generated. The precision, recall and accuracy scores are tabulated in Table 2, 3 and 4 respectively.

[2] https://www.nltk.org/

[3] https://scikit-learn.org/stable/

Label	Naive Bayes	Logistic Regression	Random Forest	SVM	ANN
Fresh	0.76	0.76	0.74	0.77	0.78
Rotten	0.74	0.74	0.75	0.77	0.77

Table 2: Precision

Label	Naive Bayes	Logistic Regression	Random Forest	SVM	ANN
Fresh	0.73	0.73	0.75	0.74	0.76
Rotten	0.77	0.77	0.74	0.78	0.78

Table 3: Recall

Accuracy	Naive Bayes	Logistic Regression	Random Forest	SVM	ANN
Accuracy	74.71 %	74.81 %	74.42 %	75.92 %	77.06 %

Table 4: Accuracy

It is observed that Artificial Neural Network is giving the best accuracy among the five models. The Precision and Recall of SVM and ANN are the highest among the given models for the two classes.

The confusion matrix for the five algorithms are plotted below:

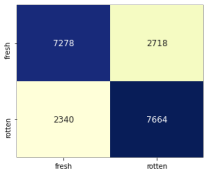

Figure 4: NB

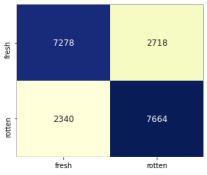

Figure 5: LR

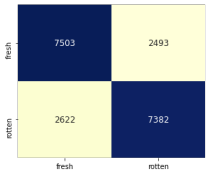

Figure 6: RF

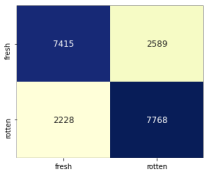

Figure 7: SVM

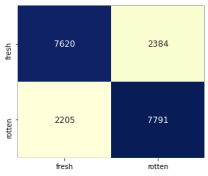

Figure 8: ANN

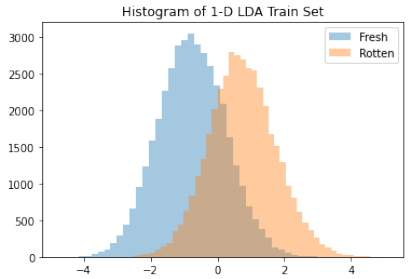

Figure 9: LDA of Train set

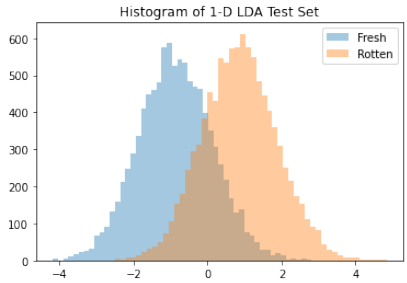

Figure 10: LDA of Test set

VI Applying LDA

After applying the five ML models on 3000 feature dataset, LDA was applied on the 3000 features. LDA was done through Sklearn's library. As LDA converts C class dataset into C-1 dimensional dataset, So our final Train set and Test set will be one dimensional (1-D). As we will not be able to clearly see the discriminative boundary on the 1-D space due to large amount of samples, we plotted a histogram of the dataset to get a clear understanding about how the labels are distributed. The plots are shown in Figure 9 and 10.

It is clear that we are not able to get a perfectly separable classes after applying LDA on the 3000 feature Train and Test set. However, we can see the distinction between the classes in Train set and Test set.

VII Results : After-LDA

After applying LDA, the five models were tested on reduced feature set. The models were tested with same set of hyperparameters applied to the original document-term matrix. The precision, recall and accuracy scores are tabulated in Table 5, 6 and 7 respectively.

It is noted that here Naive Bayes and Logistic Regression are out-performing other algorithms. There is a significant decrease in the accuracy of

Label	Naive Bayes	Logistic Regression	Random Forest	SVM	ANN
Fresh	0.75	0.75	0.71	0.75	0.74
Rotten	0.75	0.74	0.71	0.74	0.75

Table 5: Precision

Label	Naive Bayes	Logistic Regression	Random Forest	SVM	ANN
Fresh	0.74	0.74	0.70	0.73	0.75
Rotten	0.75	0.75	0.71	0.76	0.74

Table 6: Recall

Accuracy	Naive Bayes	Logistic Regression	Random Forest	SVM	ANN
Accuracy	74.58 %	74.58 %	70.90 %	74.56 %	74.57 %

Table 7: Accuracy

the Random Forest (near 3.5%).

The confusion matrix for the five algorithms are plotted below:

Figure 11: NB

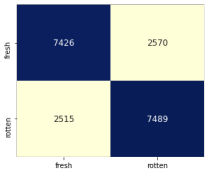

Figure 12: LR

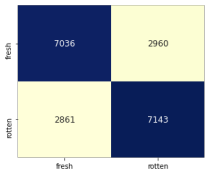

Figure 13: RF

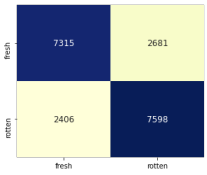

Figure 14: SVM

Figure 15: ANN

VIII Discussion

The comparison of the algorithms in terms of dimensionality reduction are tabulated below:

Algorithms	Original	After LDA	Difference
Naive Bayes	74.71	74.58	0.13 %
Logistic	74.81	74.58	0.23 %
R. Forest	74.42	70.90	3.52 %
SVM	75.92	74.56	0.36 %
ANN	77.06	74.57	2.49 %

Table 8: Differences in Test accuracy : Of Original feature dataset and LDA-applied dataset

Best accuracy in original feature space	77.06 %
Best accuracy in LDA feature space	74.58 %
Difference	2.48 %

It is observed that best accuracy in original feature matrix is obtained through Artificial Neural Network (ANN). In the LDA feature space the best accuracy is obtained through Naive Bays and Random Forest.

There is not much decrease in the accuracy of the models by transforming them from 3000 dimensional feature space to 1-D dimensional feature space (Table 8). After Conversion to 1D feature set, there is an increase in the computational speed up in the training of the five models. There is a significant reduction in the training time of the algorithms.

We can apply dimensionality reduction to the NLP applications such as Live Feed Analysis in text form. Although there exists a trade-off between the accuracy and feature set, dimensionlity reduction involves faster calculations. Feature extraction and elimination can be applied to remove redundant features in the huge document matrix

IX Conclusion

This project has implemented dimensionality reduction (LDA) on movie review dataset. Comparisons based on Test Accuracy metric were carried to analyse its effect. The analysis was done by applying five different supervised classification algorithms. It was observed that there was a loss of only 2.48 % in the accuracy by reducing 3000 feature dataset to the 1D LDA dataset. The observation states that we are able to near-achieve the level

accuracy of the original feature set by using dimensionlity reduction.

References

[1] Bolter S. Predicting product review helpfulness using machine learning and specialized classification models. 2013;.

[2] Go A, Bhayani R, Huang L. Twitter sentiment classification using distant supervision. Processing. 2009 01;150.

[3] Jurafsky D, Martin JH. Classification : Naive Bayes , Logistic Regression , Sentiment;. .

[4] Pang B, Lee L, Vaithyanathan S. Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. Association for Computational Linguistics; 2002. p. 79–86.

[5] Le QV, Mikolov T. Distributed Representations of Sentences and Documents. CoRR. 2014;abs/1405.4053. Available from: http://arxiv.org/abs/1405.4053.

[6] Taloba AI, Eisa D, Ismail SS. A Comparative Study on using Principle Component Analysis with Different Text Classifiers. arXiv preprint arXiv:180703283. 2018;.

[7] Torkkola K. Linear Discriminant Analysis in Document Classification. IEEE TextDM 2001. 2001 12;.

[8] Adlaon KM, Azcarraga J. Dimensionality Reduction of Feature Word Vectors for Sentiment Classification of Philippine Political Related Tweets; 2018. .

[9] Ljungberg BF. Dimensionality reduction for bag-of-words models: PCA vs LSA;.