SentimentAnalysis_ComparativeStudy_ResearchProposal_v7.docx

by Sanket Talaviya

Submission date: 20-Apr-2021 03:11PM (UTC+0100)
Submission ID: 149904875
File name: SentimentAnalysis_ComparativeStudy_ResearchProposal_v7.docx (1.21M)
Word count: 4328
Character count: 23803

COMPARATIVE STUDY OF SENTIMENT ANALYSIS FOR MOVIE REVIEWS

STUDENT NAME: SANKET TALAVIYA
STUDENT ID: 969198

RESEARCH PROPOSAL

LIVERPOOL JOHN MOORES UNIVERSITY – MS IN ML AND AI

DECEMBER 2020

Abstract

In today's world, Sentiment analysis has become the need of the hour for opinion mining on variety of subjects. It has various applications in marketing, e-commerce, advertising, politics and research. Day by day people expressing their sentiments over internet, different websites and social media are increasing. Sentiment analysis is the process of identifying the classifying these opinions. If one doesn't apply machine learning and AI techniques to find out overall sentiments of the particular topic going through all reviews/comments. It would take humongous effort to label this review comments manually to figure out overall sentiments. So, If we automate sentiment analysis activity using machine learning and AI, It can help saving lot of human efforts. Also, one problem in this space is that best model depends on the characteristics of the dataset model is trained on. In this research work I am going to compare different models like Logistic Regression, SVM, XGBoost, Random Forrest Ensemble and the state-of-the-art models like BERT. All these models I am going to try after applying (1) TF-IDF (2) Word2Vec Embeddings and (3) BERT Embeddings. And then using different interpretability techniques I will derive some insights for each of the models.

Table of Contents

List Of Figures.

List Of Tables

List Of Abbreviations

EDA	Exploratory Data Analysis
AUC	Area Under ROC Curve
ROC	Receiver Operating Characteristic
BOW	Bag Of Words
TF	Term Frequency
IDF	Inverse Document Frequency
BERT	Bidirectional Encoder Representations from Transformers
SVM	Support Vector Machine
XgBoost	Extreme Gradient Boosting
LIME	Local Interpretable Model-Agnostic Explanations
SHAP	Shapley Additive Explanation
KNIME	Konstanz Information Miner
TPR	True Positive Rate
FPR	False Positive Rate
TN	True Negative
FP	False Positive
TP	True Positive
FN	False Negative

1. Introduction

As Internet services are evolving more and more each day, Exponential growth has been witnessed in terms of information availability on the internet. This includes opinions and reviews for particular product or topics like movies. (Yousefpour et al., 2014). Because of tremendous value sentiment analysis brings on the table by eliminating manual efforts of parsing all reviews and figure out overall sentiment of the subject in question. Lot of researchers are attracted to this field. This area of research broadly involves NLP and Machine learning models. (Qaisar, 2020). Sentiment analysis in each field had different objectives for example if its e-commerce website reviews for a product. From the sentiment analysis product companies can gather information how good or product is doing in the market by deriving sentiments from the reviews. If the subject is movie reviews, It helps users to decide whether they should view this movie or not based on sentiment analysis that way it helps them save their time and movie if overall sentiments are negative.

Although algorithms have progressed pretty well in last few years few challenges in this field have not been completely eliminated yet. One of these challenges is to classify statements to positive or negative even though it doesn't have any clear emotions and sometimes statements can have words which have different meanings based on the contexts so that leads in to ambiguity in the end result of the classification.(Qaisar, 2020)

Film Industries including songs, trailers, movies etc rely heavily on sentiment analysis to figure out how people are feeling about particular song/trailer or movie. This sentiment analysis can also give pointers to other producers on what people don't like in movies or songs etc based on that they can take actions for their next movies. And existing movie reviews help user to decide on which movie to watch or not to watch etc and film makers in a way that what particular things people like the most and disliked the most. The dataset here is taken from Kaggle and it contains IMDB movie reviews.

In this research I am going to apply following steps on IMDB movie review dataset.

1) Retrieve IMDB dataset.
2) Pre-processing and balancing dataset.
3) Feature extraction.

4) Applying classical models like Logistic Regression, SVM, XGBoost, Random Forrest and state of the art BERT model on top of vectorization methods like TF-IDF, Word2Vec embeddings and BERT embeddings.
5) Compare performances of each model using AUC (area under ROC curve) method and confusion metrics will be shared in the results.
6) Also, I will try to extract the part of the sentence which contributed the most to derive to specific class of the sentiment.

2. Background

Sentiment analysis is useful for finding out how customers think about a particular product or topic. Before computer era started, In the earlier times sentiment analysis was done primarily on non-online modes like paper or written documents as there was no internet services. In the recent times internet services have become part of the human life and hence more and more people have started expressing their views on online forums including social medias, publications, blogs, websites etc. Sentiment Analysis is a process of extracting sentiments from these raw reviews expressed over above various internet forums using NLP and statistics. Broadly below are the two types of sentiment analysis techniques.

(1) Machine Learning based techniques: This technique mainly falls into supervised classification category, in this technique two set of datasets are required, train dataset and test dataset, Train dataset is used to learn various patterns between variables of the subject and test document is used to check the robustness of the model on the dataset that model hasn't seen yet. Logistic Regression, SVM and Naïve Bayes have achieved good performance in this area. One important decision to make in this method is to do feature selection. The most widely used features in this method are as (1) Term frequency (2) Part of Speech Information (3) Negations (4) Opinion words and phrases.

(2) Lexicon based techniques: It is unsupervised technique as it requires prior knowledge of sentiment lexicons. In this technique features of a given statements are compared with sentiment lexicons whose values are known in advance. These lexicons contain the word list that represents people's subjective feelings and opinions. Once we have these lexicons handy next task is to go through document and find out positive and negative lexicons in each statement and if statement has more positives it's a positive sentiment otherwise it's a negative sentiment. There are few methods to build sentiment lexicons (1) Manually (2) From Corpus (3) From Dictionary.

(Vohra and Teraiya, 2013)

In this research we are going to focus on Machine learning based techniques and try out different word embedding techniques with machine learning models and try finding out which works the best for movie review dataset.

3. Related Work

(Lewis, 1945) in his research did the comparative analysis of the performance of various models like Naïve Bayes, Maximum Entropy and Support Vector Machines using various features like unigram, bigram, combination of both, POS with unigrams etc. and the analysis showed that presence of the feature is more beneficial than frequency of the feature. When features are lesser Naïve Bayes works better but as it increases SVM starts doing better, Also with increased number of features Max Entropy does better than Naïve Bayes but overfits.

(Tamara and Milićević, 2018) did the comparative analysis by applying BOW, BOW with n-grams and TF-IDF embeddings to different machine learning models like Logistic Regression, SVM and few deep learning models like LSTM and ConvNets. Glove and Word2Vec embeddings were also tested here. And paper concluded that deep learning models perform better than traditional machine learning models. LSTM yielded greater than 95% accuracy and BOW n-grams + TDF with SVM model gave around 92% accuracy. BOW with Boosting yielded around 80%+ accuracy.

(Al-Saqqa and Awajan, 2019) in their survey found the in the word embedding techniques CBOW and skip-gram, Skip-gram works better for infrequent words than the other. And CBOW works well with frequent words and is faster compared to skip-gram.

(Tripathy et al., 2016) concluded in the research that accuracy of classification model is inversely proportional to the value of n in n-gram. When tests were done using higher n-gram models where n is greater than 3, Accuracy decreased for models like Maximum Entropy, SVM, NB, Stochastic Gradient De-scent(SGD). However unigrams and bigrams results were remarkably better. In this study IMDB dataset was used.

(Sahu and Ahuja, 2016) examined IMDB dataset and proposed an approach to classify sentiment expressions from zero to four on scale and extracted the features and applied ranking and used these features for training multiclass classifier to classify reviews to a correct label. They followed approach based on n-grams and after applying classification models they achieved accuracy of 88.95% as a best one.

(Tarımer et al., n.d.) conducted a study on IMDB and twitter dataset, In this study they applied Decision tree, NB, SVM machine learning algorithms on a vector space that was created in KNIME platform. That is one analytics platform. Results they obtained were showing Decision tree was performing better than NB and SVM. And when they tried for other datasets as well and finally concluded that SVM yields better results across different datasets.

(Yasen and Tedmori, 2019) In this research, applied around 8 different machine learning models after cleaning and pre-processing the dataset and they evaluated using 5 different metrics. Results concluded that Random Forrest outperformed the other classifiers and Ripper Rule Learning performed the worst on the datasets.

4. Research Questions

After all related research following research questions have been formulated.

1) Which vectorization technique works best with white box and black box models from TF-IDF, Word2Vec embeddings and BERT embeddings.
2) If these white box and black box models can be interpreted on this dataset to explain why they worked better/worse than other models.

5. Aim and Objectives

Aim of this research is to find out which vectorization technique and machine learning model works the best for analysing sentiments on movie review dataset. Here in this research we are going to apply 3 different vectorization techniques TF-IDF, Word2Vec Embeddings and BERT Embeddings to various machine learning models like Logistic Regression, SVM, Random

Forrest and BERT. Also provide interpretability for these models using various interpretability techniques at the end.

Based on the aim of the study, Following objectives have been identified.

1) Build different word embeddings using TF-IDF, Word2Vec and BERT.
2) Build predictive models using all above 3 embeddings and evaluate model's performance.
3) Compare the performance of all the models that are built and figure out which worked the best.
4) Provide interpretability for each of the models in terms of features and relevant explanations.

6. Significance of the Study

If there are some datapoints which help user decides whether movie is worth watching or not, it can really save their precious time. But now if user starts reading all reviews given on internet or social media, it will consume lot of time as it's a non-trivial task to go through all reviews manually and figure out overall sentiments around the movie. Critics today often uses social media and movie rating websites to post their comments or rating for the movies. This data can eventually help users in decision making weather to watch this particular movie or not. Through machine learning models we can automate the process of sentiment analysis from the pool of movie review comments and figure out what is the overall sentiment for the movie. This involves Natural language processing underneath and benefit that we get is incredible in terms of money and time saved. This sentiment analysis is basically a classification problem weather a particular comment is positive or negative or neutral and then for whole movie how many comments are positive and how many of them are negative, This way we can achieve overall sentiment for the movie.

7. Scope of the Study

As the time duration for this research is limited, research scope will be limited to following points:

1) The dataset used here is of IMDB movie reviews and it has been taken from Kaggle, So assuming data is already valid.

2) In this research I am going to develop and compare the performance of different machine learning models using TF-IDF, Word2Vec Embedding and BERT embeddings only.

3) Machine learning Model experiments will be limited to Logistic Regression as interpretable models and SVM, XGBoost, Random Forrest as part of black box models. Deep learning models will not be evaluated in this research or one model like BERT will be evaluated only if time permits.

4) For comparing performance of different models AUC will be used and for interpretability we are going to limit 2 methods LIME and SHAP only.

8. Research Methodology

In this section sequential steps that are going to be performed in this whole research is detailed out. This research is going to be carried out in three parts sequentially using IMDB movie review dataset. The main objective of this study is to predict whether given comments have positive or negative sentiments and possibly explain the phrase/word of the comment that contributes the most to derive to a particular classification (positive/negative) of the sentiment.

● First part entails understanding the dataset and then apply pre-processing steps, perform EDA followed by feature extraction.

● In the second part, apply create different vectors using (1) TF-IDF (2) Word2Vec Embeddings (3) BERT embeddings.

● Third part is to create models like Logistic Regression, SVM, XGBoost, Random Forrest with all these 3 embeddings and conclude the study by performing comparative study on these model's performance and derive inferences on which model worked better with some interpretability facilitator libraries like LIME and SHAP.

8.1 Dataset Description

This dataset consists of IMDB movie reviews taken directly from Kaggle Website. Dataset has 2 columns Review and Sentiment. Review column represents user reviews some of the are large comments and some of them are small. Sentiment column has two labels "Positive" and

“Negative”, Overall, there are 50000 records in the dataset and classes of positive and negative reviews are balanced that is 25000 each.
(IMDB Dataset of 50K Movie Reviews | Kaggle, 2021)

Variable	Definition
Review	User’s review statements are given as review column.
Sentiment	Classification of sentiment, i.e. positive, negative

Table1: Dataset columns overview

8.2 Data Clean up and Pre-Processing

There is an approach called Exploratory Data Analysis (EDA) that helps extracting the information enfolded in the data and based on that we can summarise the main attributes or characteristics of the data that is more important to study for a given problem.

1. Remove BR tags
2. Remove all single characters
3. Substituting multiple spaces with single space
4. Remove links
5. Remove words containing numbers
6. Converting to lowercase
7. Lemmatization
8. Feature Extractions using different embedding techniques like TF-IDF, Word2Vec Embeddings and BERT Embeddings.

8.3 Model Creation

We have created two set of data from the dataset in the 70:30 ratio where 70% data will be considered as training data and 30% will be used for testing purpose. We can apply following models on it with different embeddings and compare their performances later.

1. Logistic Regression (Whitebox)

2. SVM (Blackbox)
3. XGBoost (Blackbox)
4. Random Forrest (Blackbox)
5. BERT (Blackbox)

This IMDB movie reviews dataset has target values as positive or negative only hence it's a binary classification problem. So that's the thought process for selecting above models for classification. These all models we need to apply separately with TF-IDF, Word2Vec Embeddings and BERT Embeddings. Below is a brief summary around these methods.

TF-IDF:

Assume that we have a problem at hand where we want to figure out a which document is the most relevant for the review statement or comment that we want to search in these documents. So one naïve solution for this problem is to simply gather words of the review statement or comment and figure out how many documents do not include all the words of the review statement or comment and simply eliminate them. But since its very naïve approach intuitively we can think of that still a lot of documents will still remain relevant for the searching operation. So the next addition we can do in filtering is to count frequency of each term in the document. This is called Term Frequency.

tf(t,d) = count of t in d / number of words in d

Document frequency is similar to term frequency, It represents the count of a particular term in the set of document. In other words, It is the number of documents where the given term is present.

df(t) = occurrence of t in documents

In the term frequency, all the terms are given equal importance or weightage. But in practical scenarios some words occur too frequently like "is", "are", "of" etc. and that has very less to no importance, hence we need to rationalise or weigh down high frequency terms and weigh up the low frequency terms, IDF is an factor which helps achieving this by lowering weightage of high frequency terms and increasing weightage for low frequency terms.

$$idf(t) = N/df$$

Here N is total number of documents and DF is number of documents containing given term.

Term	Review 1	Review 2	Review 3	IDF	TF-IDF (Review 1)	TF-IDF (Review 2)	TF-IDF (Review 3)
This	1	1	1	0.00	0.000	0.000	0.000
movie	1	1	1	0.00	0.000	0.000	0.000
is	1	2	1	0.00	0.000	0.000	0.000
very	1	0	0	0.48	0.068	0.000	0.000
scary	1	1	0	0.18	0.025	0.022	0.000
and	1	1	1	0.00	0.000	0.000	0.000
long	1	0	0	0.48	0.068	0.000	0.000
not	0	1	0	0.48	0.000	0.060	0.000
slow	0	1	0	0.48	0.000	0.060	0.000
spooky	0	0	1	0.48	0.000	0.000	0.080
good	0	0	1	0.48	0.000	0.000	0.080

- Review 1: This movie is very scary and long
- Review 2: This movie is not scary and is slow
- Review 3: This movie is spooky and good

Figure 1: TF-IDF Representation

(BoW Model and TF-IDF For Creating Feature From Text, 2021)

Word2Vec Embeddings:

Word vector helps representing the words as vectors. So you can vectorize these words and match these vectors to find similarities, But this sounds impractical if you have millions of words. So we might need to find similarities between these encoded vectors.

Earlier N-Grams were used heavily to extract the meaning of the word with surrounded other words but it can't capture the context.

Based on the different contexts the words are found, Word Embeddings can be defined as high-dimensional vectorised view of words. Word embeddings helps comparing the similarity of words and provide some useful info for machine learning models to consider as input.

(Introduction to Word Vectors and Word2Vec - Studytonight, 2021)

Word2Vec is very well known predictive model that strives to output a target word provided a context or the context words from the target. It's a vector representation of words such that the cosine distance between those vectors reflects the semantic "distance" between the words they represent. For example if word1 and word2 have a similar meaning, then cos distance of word1 vector and word2 vector should be small.

Word embeddings help achieving dimensionality reduction so number of feature required to build the model is less. Also it captures the meanings of the words and semantic relationships as well as context.

Source Text	Training Samples generated from source text
I will have orange juice and eggs for breakfast	(will, I) (will, have) (will, orange)
I will have orange juice and eggs for breakfast	(have, I) (have, will) (have, orange) (have, juice)
I will have orange juice and eggs for breakfast	(orange, will) (orange, have) (orange, juice) (orange, and)
I will have orange juice and eggs for breakfast	(juice, have) (juice, orange) (juice, and) (juice, eggs)
I will have orange juice and eggs for breakfast	(and, orange) (and, juice) (and, eggs) (and, for)
I will have orange juice and eggs for breakfast	(eggs, juice) (eggs, and) (eggs, for) (eggs, breakfast)
I will have orange juice and eggs for breakfast	(for, and) (for, eggs) (for, breakfast)

The word highlighted in yellow is the source word and the words highlighted in green are its neighboring words.

Figure 2: Skip Gram Representation

(NLP 101: Word2Vec — Skip-gram and CBOW | by Ria Kulshrestha | Towards Data Science, 2021)

BERT Embeddings:

DistilBERT is a lightweight version of BERT with comparable accuracy. DistilBERT processes the sentence and passes along some information it extracted from it on to the next model that is Logistic regression/SVM/Random Forrest model from scikit learn, It will take the result of DistilBERT's processing, and classify the sentence.

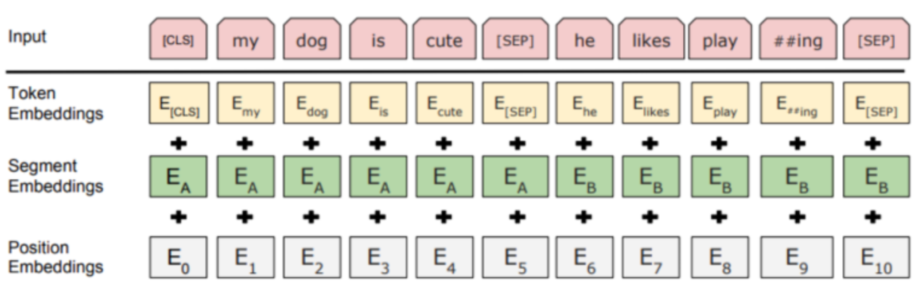


Figure 3: BERT embeddings

(What is BERT | BERT For Text Classification, 2021)

BERT Embeddings comprises of 3 different embeddings.

1) Position Embeddings are used to understand the location index information of the input. This can overcome the shortcomings of RNN which doesn't keep location information.

2) Segment Embeddings are unique embeddings for different sentences that is learnt by BERT. In these embedding different sentences are given different embeddings.

3) Token Embeddings are unique embeddings in which BERT learn specific token from the word token vocabulary.

8.4 Research Methodology Flow Chart

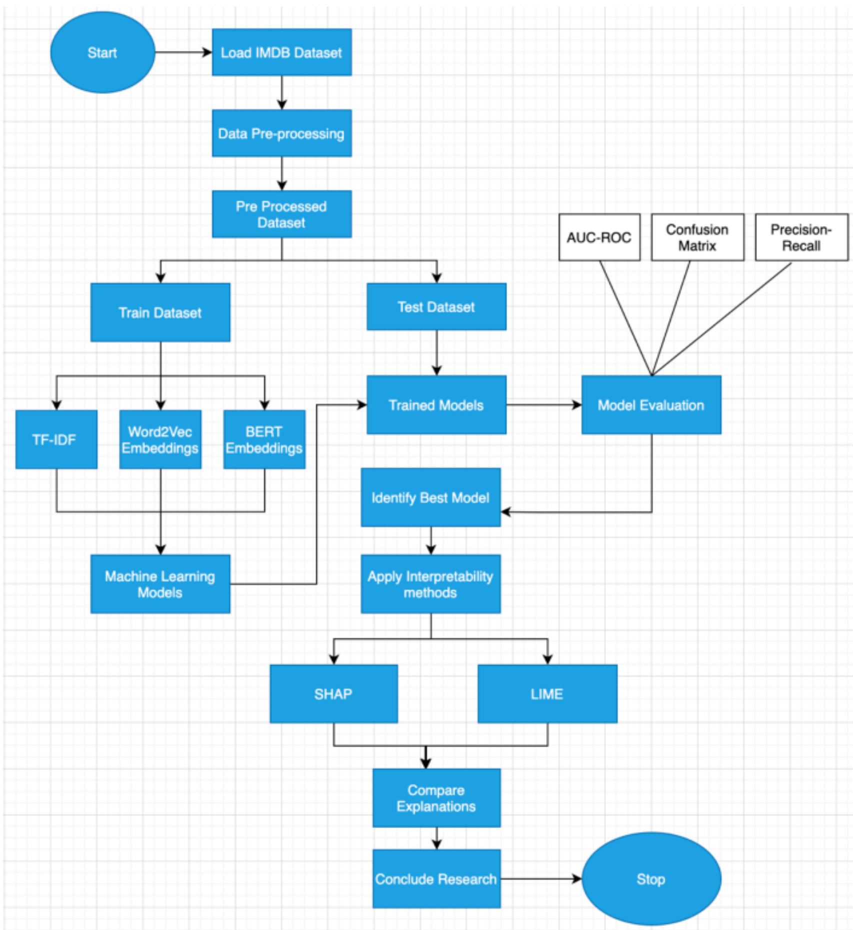


Figure 4: Research Methodology Flow-Chart

8.5 Model Evaluation

Since it's a binary classification problem we are going to use AUC (Area Under Curve) – ROC (Receiver Operating Characteristics) as a metric to evaluate various model performances.
By calculating AUC-ROC, you can determine the goodness of the model. If the curve is sticky towards upper left side, it means that the model is very good and if it is more towards the 45-degree diagonal, it means that the model is almost completely random. So, the larger the AUC, the better will be your model. So In the best case it passes through the upper left corner of the graph. So the least area that an ROC curve can have is 0.5, and the highest area it can have is 1.

ROC Curves shows the trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR)

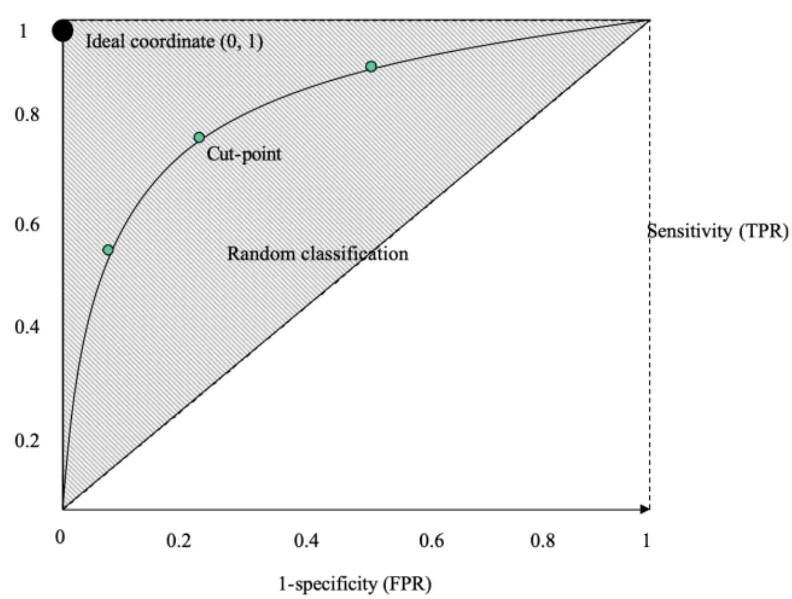


Figure-5: AUC-ROC Curve

(Zhu et al., 2010)

Terms used in AUC-ROC Curve

1. TPR can be defined as below

$$TPR = \frac{\text{True Positive}}{\text{True Positive + False Negative}}$$

2. Specificity can be defined as below.

$$\text{Specificity} = \frac{\text{TN}}{(\text{TN} + \text{FP})}$$

3. FPR can be defined as below.

$$\text{FPR} = \frac{\text{False Positive}}{\text{False Positive} + \text{True Negative}}$$

So for the best model we can expect high TPR and low FPR and depending on the study we can decide if high precision is more important or high recall.

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{\text{TP}}{\text{TP} + \frac{1}{2}(\text{FP} + \text{FN})}$$

8.6 Interpretability of the models

1) **LIME**: LIME is an algorithm that helps explaining predictions of any classification or regression model by approximating it locally with some interpretable model. LIME outputs set of interpretations that helps interpreting contribution of each variable or feature to the prediction for a given input sample. This is in form of local interpretability. Linear Regressions, Decision Trees etc. are used as interpretable models with this algorithm and these models are trained with small augmentation so to the original data to achieve good local approximation. (LIME Explained | Papers With Code, 2021)

2) **SHAP**: This is a technique with strong support from game theory that helps explaining the predictions of any Machine Learning model. It helps deriving local variable contributions to overall prediction. It presents essentially three properties 1. Local accuracy 2. Missingness 3. Consistency. (5.10 SHAP (SHapley Additive exPlanations) | Interpretable Machine Learning, 2021)

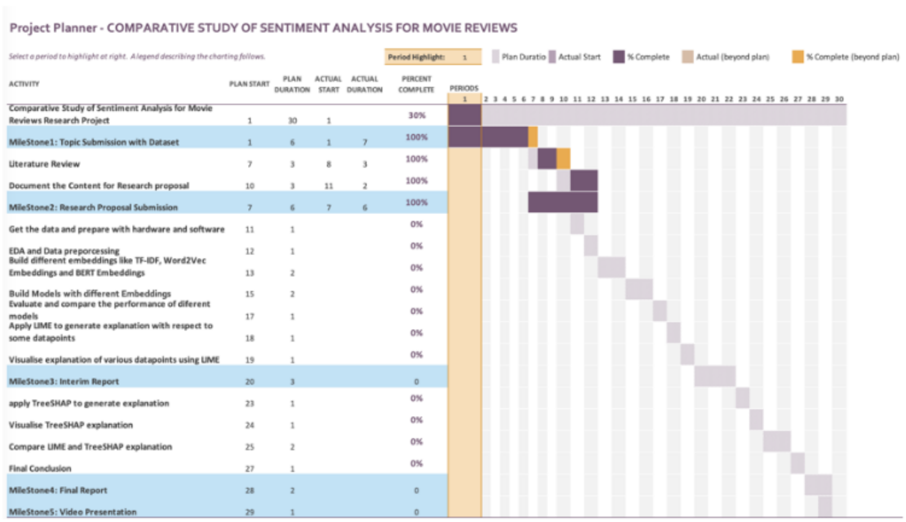
8.7 Results

The research will be concluded based on comparative study of the performance of the above models with all embeddings mentioned(TF-IDF, Word2Vec Embeddings, BERT Embeddings). Performances of the models will be compared based on AUC-ROC primarily. I will also try to achieve interpretability of classical model using techniques like LIME and SHAP. Then I will compare the insights drawn from LIME and SHAP both.

9. Required Resources

1. Hardware Requirements
    1. Operating System: MacOS BigSur.
    2. Processor: 2.2 GHz 6-Core Intel Core i7
    3. Memory: 16 GB
2. Software Requirements
    1. Python Version: 3.3
    2. Jupyter Lab: 2.2.6
    3. Python Libraries: Pandas, NumPy, Matplotlib, Seaborn, scikit-learn.
    4. Some interpretation libraries available in python like LIME, SHAP etc
    5. Access to cloud offerings like Kaggle notebooks, Google Collab etc.

10. Research Plan



References

Al-Saqqa, S. and Awajan, A., (2019) The Use of Word2vec Model in Sentiment Analysis: A Survey. *ACM International Conference Proceeding Series*, June 2020, pp.39–43.

Anon (2021) *5.10 SHAP (SHapley Additive exPlanations) | Interpretable Machine Learning*. [online] Available at: https://christophm.github.io/interpretable-ml-book/shap.html#definition [Accessed 11 Apr. 2021].

Anon (2021) *BoW Model and TF-IDF For Creating Feature From Text*. [online] Available at: https://www.analyticsvidhya.com/blog/2020/02/quick-introduction-bag-of-words-bow-tf-idf/ [Accessed 11 Apr. 2021].

Anon (2021) *IMDB Dataset of 50K Movie Reviews | Kaggle*. [online] Available at: https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews [Accessed 9 Apr. 2021].

Anon (2021) *Introduction to Word Vectors and Word2Vec - Studytonight*. [online] Available at: https://www.studytonight.com/post/introduction-to-word-vectors-and-word2vec [Accessed 17 Apr. 2021].

Anon (2021) *LIME Explained | Papers With Code*. [online] Available at: https://paperswithcode.com/method/lime [Accessed 11 Apr. 2021].

Anon (2021) *NLP 101: Word2Vec — Skip-gram and CBOW | by Ria Kulshrestha | Towards Data Science*. [online] Available at: https://towardsdatascience.com/nlp-101-word2vec-skip-gram-and-cbow-93512ee24314 [Accessed 11 Apr. 2021].

Anon (2021) *What is BERT | BERT For Text Classification*. [online] Available at: https://www.analyticsvidhya.com/blog/2019/09/demystifying-bert-groundbreaking-nlp-framework/ [Accessed 11 Apr. 2021].

Dholpuria, T., Rana, Y.K. and Agrawal, C., (2018) A sentiment analysis approach through deep learning for a movie review. *Proceedings - 2018 8th International Conference on Communication Systems and Network Technologies, CSNT 2018*, pp.173–181.

Haque, M.R., Akter Lima, S. and Mishu, S.Z., (2019) Performance Analysis of Different Neural Networks for Sentiment Analysis on IMDb Movie Reviews. *3rd International Conference on Electrical, Computer and Telecommunication Engineering, ICECTE 2019*, pp.161–164.

Lewis, S.J., (1945) Thumbs up. *American Journal of Orthodontics and Oral Surgery*, 319, pp.481–482.

Qaisar, S.M., (2020) Sentiment Analysis of IMDb Movie Reviews Using Long Short-Term Memory. *2020 2nd International Conference on Computer and Information Sciences, ICCIS 2020*, pp.7–10.

Sahu, T.P. and Ahuja, S., (2016) Sentiment analysis of movie reviews: A study on feature selection and classification algorithms. *International Conference on Microelectronics, Computing and Communication, MicroCom 2016*, pp.0–5.

Tamara, K. and Milićević, N., (2018) Comparing Sentiment Analysis and Document Representation Methods of Amazon Reviews. *SISY 2018 - IEEE 16th International Symposium on Intelligent Systems and Informatics, Proceedings*, pp.283–288.

Tarımer, İ., Çoban, A. and Kocaman, A.E., (n.d.) *Sentiment Analysis on IMDB Movie Comments and Twitter Data by Machine Learning and Vector Space Techniques*.

Tripathy, A., Agrawal, A. and Rath, S.K., (2016) Classification of sentiment reviews using n-gram machine learning approach. *Expert Systems with Applications*, [online] 57March, pp.117–126. Available at: http://dx.doi.org/10.1016/j.eswa.2016.03.028.

Vohra, M.S.M. and Teraiya, P.J.B., (2013) Journal of Information, Knowledge and Research in Computer Engineering a Comparative Study of Sentiment Analysis Techniques. *Journal of Information,Knowledge and Research in Computer Engineering*, pp.313–317.

Yasen, M. and Tedmori, S., (2019) Movies reviews sentiment analysis and classification. *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology, JEEIT 2019 - Proceedings*, pp.860–865.

Yousefpour, A., Ibrahim, R., Hamed, H.N.A. and Hajmohammadi, M.S., (2014) A comparative study on sentiment analysis. *Advances in Environmental Biology*, 813, pp.53–68.

Zhu, W., Zeng, N. and Wang, N., (2010) *Sensitivity, Specificity, Accuracy, Associated Confidence Interval and ROC Analysis with Practical SAS ® Implementations*.

SentimentAnalysis_ComparativeStudy_ResearchProposal_v7....

ORIGINALITY REPORT

2%	1%	0%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Liverpool John Moores University Student Paper	2%

Exclude quotes	Off	Exclude matches	< 1%
Exclude bibliography	On		