Youtube Comments Sentiment Analysis

Shashi Kant[1]
Dept. of Computer Science and Engg.
Sharda University
Greater Noida, India
immnnitian@gmail.com

Raushan Kumar Upadhyay[2]
Dept. of Computer Science and Engg.
Greater Noida Institute of Technology
Noida, India
raushanupadhyay011@gmail.com

Palak Gupta[3]
Dept. of Computer Science and Engg.
Greater Noida Institute of Technology
Noida, India
palak.g882@gmail.com

Raghav Tiwari[4]
Dept. of Computer Science and Engg.
Greater Noida Institute of Technology
Noida, India
raghav312002@gmail.com

Honey Kumar[5]
Dept. of Computer Science and Engg.
Greater Noida Institute of Technology
Noida, India
honeykumar8642@gmail.com

2024 1st International Conference on Advances in Computing, Communication and Networking (ICAC2N) | 979-8-3503-5681-6/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICAC2N63387.2024.10895870

Abstract— Over the years, there has been a significant surge in textual information, leading to a burgeoning research interest. The contemporary focus lies in the intriguing field of sentiment analysis on YouTube comments. Despite the substantial volume of user comments and reviews on many videos, limited efforts have been directed towards extracting meaningful trends due to the inherent inconsistency and variable quality of information.

In this study, we perform sentiment analysis on YouTube comments concerning popular topics, with a selection of various machine learning techniques and algorithms. The aim will be to highlight how sentiment analysis can show trends, seasonality, and forecasts, and offer an insightful view into the effects of real-world events on the public mood. The results display strong correlation views between users' sentiments toward corresponding keywords and the real-world events.

The primary aim of this research is to assist scholars in identifying high-quality research papers on sentiment analysis. Our approach involves sentiment analysis of YouTube comments using citation sentences based on an existing annotated corpus consisting of 1500 citation sentences. Data cleansing involved the application of various normalization rules to eliminate noise from the comments in the corpus.

Keywords - Sentimental analysis; citations; machine learning; classification;

1. INTRODUCTION

The research focuses on gathering YouTube comments made by the public with the aim of evaluating user attitudes toward different aspects of a video expressed in their written descriptions. Sentiment analysis is important, as such analyses allow for the quick and clear understanding of the sentiments expressed on a large volume of text, thus translating user opinions into actionable value. This generally includes positive, negative, and neutral sentiments, views, attitudes, impressions, emotions, and feelings expressed in the text.

As one of the leading and most widely used platforms, YouTube sustains staggering figures: over 1 billion unique users that watch more than 6 billion hours of video per month. It accounts for 20% of web traffic and 10% of total internet traffic. YouTube has a number of social mechanisms for gauging user feedback, such as voting, rating, favoriting, sharing, and commenting. Particularly notable is the fact that Youtube does more than merely share videos: users can subscribe to video channels, comment on videos, and interact with one another, resulting in a fertile mix of implicit and explicit interactions among users. It is this social dimension of YouTube that separates it from its more traditional content providers.

This study conducts sentient analysis over a public comment stream from the annotated corpus of 1500 citation sentences. Based on mapping rules, the annotation of the corpus where polarity is assigned to citation sentences was done accordingly.

2. RELATED WORK

Studies conducted by different researchers on sentiment analysis cover Twitter and YouTube. They perform the analysis of comments, tweets, and metadata picked from user profiles or public events associated with these social networks in order to derive essential observations about their overall usage patterns.

Still, sentiment classification accuracy will not be as high as that for standard topic-based text categorization via machine learning methods. The arising difficulty is the concurrent presence of some positive and negative expressions in the review, thus complicating accurate emotion prediction.

In a related study on YouTube comments, Smita Shree and Josh Brolin proposed a lexicon-based, unsupervised approach to sentiment polarity detection. In spite of a knowledge-based approach and the preparation of a social media lexicon representing user sentiments, the results indicated a lower recall rate for negative sentiments than for positive

ISBN: 979-8-3503-5681-6/24/$31.00©2024 IEEE

ones, blaming this on a wider variability in the linguistic expression of frustration and dissatisfaction.

Other research around sentiment analysis carried out in social networks like Twitter emphasizes moods of individuals and how they are influenced by events in the social, political, cultural, and economic arena. A very recent work by A. Kowcika et al. explores how to extract information from Twitter and to conduct efficient sentiment analysis on tweets related to the Smartphone war aiming to classify users based on age and gender. Krisztian Balog et al. also proposed a method for sentiment analysis about the Smartphone war using a very effective rating process.

The paper "Twitter Sentiment Analysis: the Good, the Bad, and the OMG!" authored by Efthymios Kouloumpis et al. describes a study in which the effectiveness of language features in detecting Twitter sentiment is analyzed. The existing lexical resources and some relating informal language were evaluated in doing this. Also, sentiment analysis of blog posts was also done by Gilad Mishne et al.

Daniele Riboni performed significant work in the classification of Web pages, concerning feature selection in the paper "Feature Selection for Website Classification." Experimenting on a corpus of 8000 documents from 10 Yahoo! categories, Kernel Perception and Naive Bayes classifiers were utilized. The study pointed out how dimensionality reduction is important and a new structured weighing procedure is presented. A new representation of linked pages through local information was also introduced, making hypertext categorization appropriate for real-time applications.

Another pertinent classification study, conducted by Eibe Frank et al., proposed an effective correction by adjusting attribute priors. As this adjustment was installed as an additional normalization step in the data, hence it strengthened the area under the ROC curve. The authors demonstrated the close relationship between the modified version of MNB and the simple centroid-based classifier, comparing the two methods empirically.

Diana Maynard et al. investigated a multimodal approach in their paper on social media sentiment analysis. The goal was to help archivists select material for an archive about social media while emphasizing a structured preservation on semantic categories. The rule-based textual approach accounted for challenges unique to social media, including fact-finding with noisy grammatical text alongside expletives and sarcasm.

References consist of Athar, Pang, Lee, Vaithyanathan, Liu, Pinkesh Badjatiya et al., and Mehmood, Essam, Shafi. They appear to examine different aspects of sentiment analysis, opinion mining, deep learning for hate speech detection, and sentiment analysis in Roman Urdu, contributing some more insights to the respective domain of research.

3. METHODOLOGY

This section defines the purpose and procedure outlined in Fig. 1. It initiates with an annotated dataset and is based on Scikit-Learn, a Python machine learning library, very famous for compatibility with Python and its ease of use.

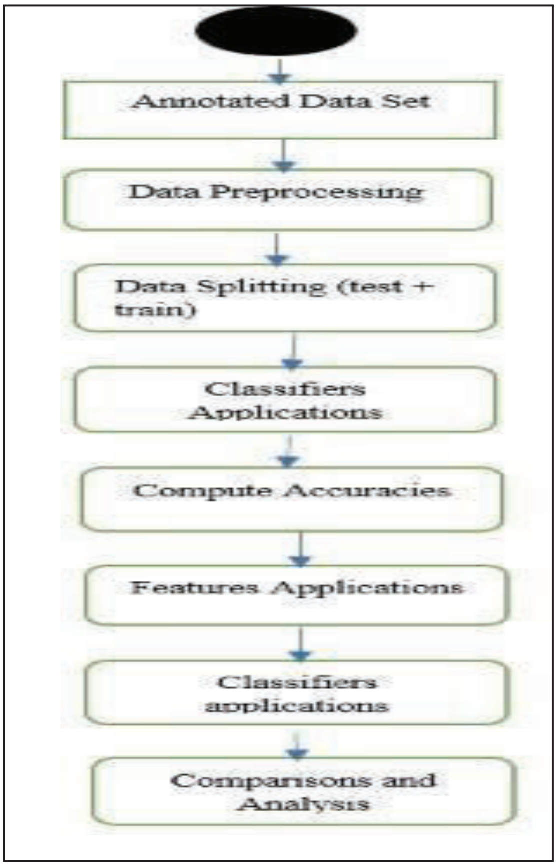
The system reads data from a TSV-formatted file. After that, data is put through a pre-processing stage to clean and prepare the data so that machine learning algorithms can act on it. Since machine learning algorithms require numeric input, textual data gets converted to a suitable format using the "count-vectorizer" Scikit -Learn module, yielding a matrix of token counts. Finally, the data is split randomly with 60% being used to train the classifier and 40% is used to test the classifier's accuracy.

Two phases of experimentation are performed: N-grams (Length 1-3) features are extracted, and an accuracy measure using both accuracy score and F-score is computed in the first phase. Phase 2 incorporated several additional features: stop words and punctuation removal, lemmatization along with the N-grams-based feature, all of which serve the purpose of minimizing noise and complexity, thus maximizing the accuracy of the system. The experiments are repeated for thirty iterations with their average result taken into account, making up a total of six experiments where accuracies are computed for both phases. Feature engineering is vastly important for contextual representation, especially N-grams features (Length 1-3). Several features, like stop and punctuation removal, were incorporated to reduce noise and increase complexity throughout the data set. In training and testing the classifiers, we used machine learning algorithms for sentiment analysis. The training part discussed was about the parameter selections and any tunings performed to optimize classifier performance. Evaluation metrics beyond F-score and accuracy were used in the assessments to provide a complete picture of the classifier's effectiveness.

The procedure of splitting of data for training and testing purposes was elaborated upon, emphasizing the randomness of the process to get a representative sample. Thirty iterations were taken on each experiment to substantiate the results. The experimental design was justified, pointing to the chosen features and how they influence sentiment accuracy. Control variables and constant variables were considered as part of the research, providing an almost constant environment for the experiment.

Feature engineering was crucial, wherein N-grams features (Length 1-3) were selected for easy contextual information extraction. To reduce noise while building vocabulary complexity, features like removing stop words and punctuation were identified. Machine learning algorithms used for sentiment analysis were classified and tested in training. It was specified how the training was done, what parameters were put into use, and what tuning was performed to enhance classifier performance. Besides F-score and accuracy, a few other metric evaluations were incorporated in order to provide a holistic assessment of the classifier's effectiveness.

Where applicable, some references were made to statistical analysis to endorse the significance of observed differences or improvements in correctness. Information on the computational resources used in the conduct of the experiments was mentioned for good faith in the experimental configuration.



3.1 EVALUATION METRICS

Above all, an assessment of any research is a statement on its standing and quality. This chapter briefly describes the metrics that this study utilizes to assess the performance of sentiment analysis systems. Indeed, performance is gauged through the computation of classification accuracy F-score and accuracy score for sentiment analysis systems.

A False Positive has been defined as a type-1 error or false positive case, while a False Negative has been defined as a type-2 error or false negative case. The common metric is F-score, and it provides an overall indication of performance by acting as a harmonic mean of precision and recall.

3.2 DATA PRE-PROCESSING

The corpus for sentiment analysis classification was created by preparing a dataset made up of 1500 citation sentences labeled positively, negatively, and neutrally. The classifier was then trained on some of these sentences, which had been selected randomly in a ratio of 60:40 in some arbitrary way, after cleaning the data thoroughly to enhance the accuracy of the system and after the application of predefined rules.

A .Features Selection

To build a successful sentiment analysis system, various features have been used by different machine learning model inputs. Such features include lemmatisms, n-grams, stop words, and term-document frequency to assess the accuracy of the classifier, with the complete evaluation results presented later.

B. Lemmatization

The process of lemmatization was used to overcome the problem of ambiguity in homographic words and in inflectional word forms. Some examples of inflected forms of the word "Talk" are "Talking," "Talks," and "Talked." We have chosen lemmatization over stemming because coming up with rules for their use is not easy. Both have basically both pros and cons, but stemming is more suited for shorter retrieval lists, while our application, which is working with huge batches of input lists, found lemmatization to be much more effective. Differences in how derivations are actually represented have not inhibited lemmatization from achieving any normalizing function upon inflected forms of the word.

C. Stop Words and Punctuation

In English texts, there is an abundance of meaningless and non-informative words known as stop words. The stop-word removal technique was adopted to rid the data of certain decorations that would otherwise interfere in classification and increase the size of the data set. This idea is in line with other studies that have suggested the removal of stop words to reduce data dimensions.

3.3 ALGORITHMS USED

This work attempted to utilize six machine learning techniques for the task of sentiment analysis. The modelling of all techniques is briefly discussed below.

Classification:-

Once the pre-processing and feature selection was done, a plethora of text classifiers, suggested in the literature, can be applied. In this paper, six discrimination algorithms were employed:

i. Naïve-Bayes:

Naïve-Bayes is a commonly and widely used classification algorithm known for its simplicity and efficacy. The classifier works on the basis of Bayes' theorem , using the concept of probabilities on the level of classification. Naïve-Bayes is very favorable toward text classification because it requires smaller data sets for training. These steps include the removal of numeric, foreign words, HTML tags, and special symbols such that a set of words remain afresh. This pre-processing gives the pair word-category to the training samples.

Consider a word 'y' from the test set (unlabeled word set) and a window of n-words (x1, x2, …xn) from a document. Based on the Bayes theorem, the conditional probability of 'y' based on n-words from the training set may be determined. Because of this simplicity and the requirement for little data to train, it has been proven to be an effective technique in text classification.

The conditional probability of given data point ‘y’ to be in the category of n-words from training set is given by:

$$P(y/x1,x2,\cdots\cdots xn)=P(y)\times\prod i=1nP(xi/y)P(x1,x2,\cdots\cdots xn)$$

ii. Support Vector Machine (SVM):

In machine learning, SVMs may be considered as a strong supervised learning algorithm that has yielded substantial improvements in various tasks of discourse, especially sentiment analysis. SVM classifiers are strong at distinguishing complex data, giving excellent predictions under increasing complexities of data.

iii. Decision Tree:

ISBN: 979-8-3503-5681-6/24/$31.00©2024 IEEE

The Decision Tree classifier is commonly used for text classification due to its fascinating way of creating classification rules. NLP researchers are drawn to it because there are just stimulated using data from the dataset to form decision trees. The advantages of decision trees include creating understandable prediction rules, constructing the shortest and fastest tree, and requiring only as many features as there are to classify all the data. Decision trees, however, face challenges such as overfitting, testing one attribute at a time before decision-making, and inability to cope with numerical attributes and missing values. Hyperparameters of Decision Trees including max_features, min_sample_split, and max_depth, etc., are optimized to reduce overfitting.

iv. Random Forest:

Random Forest, an all-round classifier, is noticed for its efficiency and discriminative classification capabilities. The performance of this algorithm stands out against other classifiers, marking it as an interesting and efficient choice.

v. K-th Nearest Neighbour (KNN):

K-th Nearest Neighbour (KNN) is noted as a simple and effective classifier, a "lazy learner," because there is no real training phase, save for keeping all training examples in memory. KNN is very efficient, but during the storing of training values, much memory space is required. The algorithm identifies a given value of K for the K nearest neighbours of an unobserved data point and assigns to it the class of the majority class of the K neighbours.

3.4 FEATURES USED

Word clouds or tag clouds are the graphical representations of how frequently words are used, giving prominence to the words, which come frequently in a particular source text. The visualization shows the words more prominently in the comments as they occur more frequently, whereby the size of each word corresponds to its frequency. This visualization offers evaluators some value in exploratory text analysis, capturing the frequent words in a set of interviews, documents, or such text. Word clouds can also successfully communicate key points or themes during reporting.

4. RESULT

There were various machine learning algorithms employed for classification and the efficiency of the systems was assessed accordingly with the help of the evaluation metrics. A thorough description of the experimental results obtained with the written metrics is shown in Tables I and II.

Logistic Regression would emerge as the best overall performer in micro-average, even without the addition of extra features. Use of unigrams greatly enhanced the performance of Logistic Regression and Decision Trees. Furthermore, the combination of unigrams with other features greatly enhances Naive Bayes, k-Nearest Neighbors, and Random Forest performance. Decision Trees quite consistently outperform other models in reference to unigrams, bigrams, and trigrams.

Logistic Regression (LR) demonstrates superior performance in the micro-average, even without the incorporation of additional features. The performance of both Logistic Regression (LR) and Decision Trees (DT) is improved when unigrams are introduced. Beyond that, when unigrams are combined with other features, Naive Bayes (NB), k-Nearest Neighbors (KNN), and Random Forest (RF) show significant performance increases. DT performs best within unigrams, bigrams, and trigrams.

LR performs admirably with unigrams, while k-th nearest neighbour does similarly with n-grams, B knowing it to have been remarkable in full instead with other features in addition. Similarly, it is noticed that RF works just like KNN, with optimal results under feature expansion.

LR shall be gratified for a lion's share of the credit when it comes to fruitful unigrams-only performance. The same goes awry for the k-th nearest neighbour's expectations in this n-grams-game-when other features come in handy-by giving RF a run for KNN's money.

From every angle, it appears that best performance is yielded on using unigrams, bigrams, and trigrams alone without feature enhancements, given that unigrams rank the highest. An examination of Table II shows SVM, LR, and RF among the most productive yielding the highest accuracy scores.

5. CONCLUSION

The sentiment analysis system is built on YouTube comments by employing different machine learning classifiers such as Naïve Bayes, support vector machine, decision tree, logistic regression, k-nearest neighbor, and random forest. Different features are used to go through the classification and optimization, applying his dataset that is divided into training and testing sets in action. The performance of classifiers was evaluated through different metrics including F-score and accuracy.

As much as using the uni-gram helps achieve a micro-F score of 87%, an enhancement yielding a 9% improvement over the base system. Further enhancing by lemmatization and removal of stop words yields a macro-F score of 49%. Long story short, the system achieves consistent micro-F scores at 87% using bi-gram and tri-gram features, optimizing itself further to an 11% with additional pairwise training.

We got this number for the macro-F to be 49% through a reduction in the dimensions of data by means of the lemmatization process and a stop-word removal mechanism. In conclusion, we have made very reasonable progress in gaining a maximum micro-F of 87% and macro-F of 49%, not underscoring the considerable advances we have done in sentiment analysis on YouTube comments.

5. REFERENCES

[1] Athar, A. (2014). Sentiment analysis of scientific citations (No. UCAMCL-TR-856). University of Cambridge, Computer Laboratory.

[2] Athar, A., Teufel, S. (2012, July). Detection of implicit citations for sentiment detection of the Workshop on Detecting Structure in Scholarly Discourse (ppp. 18-26).

[3] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... &Vanderplas J. (2011). Scikit-learn: Machine learning in Python. Journal of machine learning research.

[4] Poria S, Cambria E, Gelbukh A, Bisio F, Hussain A (2015) Sentiment data flow analysis by means of dynamic linguistic patterns

[5] Turney PD, Mohammad SM (2014) Experiments with three approaches to recognizing lexical entailment.

ISBN: 979-8-3503-5681-6/24/$31.00©2024 IEEE

[6] Parvathy G, Bindhu JS (2016) A probabilistic generative model for mining cybercriminal network from online social media: a review.

[7] Qazvinian, V., &Radev, D. R. (2010, July). Identifying non-explicit citing sentences for citation-based summarization. In Proceedings of the 48th annual meeting of the association for computational linguistics (pp. 555-564). Association for Computational Linguistics.

[8] Socher R (2016) deep learning for sentiment analysis—invited talk. In: Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis.

[9] Sobhani P, Mohammad S, Kiritchenko S (2016) Detecting stance in tweets and analyzing its interaction with sentiment. In: Proceedings of the 5th joint conference on lexical and computational semantics.

[10] Saif, H., He, Y., & Alani, H. (2012, November). Semantic sentiment analysis of twitter. In international semantic web conference (pp. 508- 524). Springer, Berlin, Heidelberg.

[11] Dashtipour K, Poria S, Hussain A, Cambria E, Hawalah AY, Gelbukh A, Zhou Q (2016) Multilingual sentiment analysis: state of the art and independent comparison of techniques.

[12] Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore. Twitter Sentiment Analysis: The Good the Bad and the OMG! In Proceedings of the Fifth International Conference on Weblogs and Social Media, ICWSM, (2011).

[13] Cambria E, White B (2014) Jumping NLP curves: a review of natural language processing research.

[14] Mohammad SM, Zhu X, Kiritchenko S, Martin J (2015) Sentiment, emotion, purpose, and style in electoral tweets.

[15] G. Eason, B. Noble, and I.N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955. (references)

[16] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

ISBN: 979-8-3503-5681-6/24/$31.00©2024 IEEE