12th International Conference on Computational Intelligence and Communication Networks

Analyzing Sentiment using IMDb Dataset

Sandesh Tripathi, Ritu Mehrotra, Vidushi Bansal, Shweta Upadhyay
Department of Computer Science
Birla Institute of Applied Sciences, Bhimtal, India
E-mail: tripathisandesh@birlainstitute.co.in, mehrotra@birlainstitute.co.in, vidushibansal@birlainstitute.co.in, shwetaupadhyay@birlainstitute.co.in

ABSTRACT- Text is the largest repository of human knowledge acquired over thousands of years. This knowledge will impart even more meaning if mined for deeper insights. Sentiment Analysis (SA) provides a traditional machine learning (ML) solution to this problem by putting Natural Language Processing (NLP) to work. In the proposed work, we have performed SA on the IMDb movie reviews dataset taken from Kaggle's Bag of Words meets Bag of Popcorn challenge to demonstrate how valuable insights can be drawn from a bulk of textual data collected from the internet. We derive these insights by applying four traditional ML algorithms namely, Naïve Bayes (NB), Logistic Regression (LR), Random Forest (RF), and Decision Tree (DT). Furthermore, results of these four algorithms were compared on the basis of six evaluation metrics – confusion matrix, accuracy, precision, recall, F1 measure, and Area Under Curve (AUC).

Keywords- Sentiment Analysis; IMDb reviews dataset; Bag of Words; Logistic Regression; Naive Bayes; Decision Tree; Random Forest.

I. INTRODUCTION

The main purpose of sentiment analysis (SA) is to detect the polarity of a statement that is to be inspected. The polarity can result in either a positive, negative or a neutral value. It is important to know the meaning of the statement to obtain the implication of the sentence and to point to the node of polarity where the statement is referring to. This includes the understanding of text processing and analysis. While this becomes easy for a person to reach the conclusion, the machine tends to take slow steps to the outcomes. In places where the number of tweets increases and analysing documents are large, it becomes impossible for a single person to evaluate the sentiment of the texts. Thus, we require the ability of machines to process the data in a short period of time and generate the results. With the ability to learn and process things at a higher rate, the machines never fail to produce the best result.

The knowledge of the sentiments of the people is very important for business purposes as it serves as a base to the needs and requirements of the customer upon which the business can produce good quality goods. The feedback of the customers is equally important as it provides valuable insights governing the tendency to like or dislike a product. This way the demand of the project will be met by the organisations. It will also help the business to know the performance of their project in the market and check if the customers are satisfied with the quality and pricing plans.

Text processing can be very difficult as it includes the analysis of streams of data of reviews. This takes a lot of time and effort from the people. The data can be structured, semi-structured or unstructured depending upon the repository and platform. Analysis of this dispersed data is possible for machines by the use of capturing the tag words that may relate to the actual meaning of the entire sentence. This way, the tokens can be registered describing the polarity of the statement. It can perform real time analysis delivering results efficiently. Sentiment analysis can be performed in Rule-based, Automatic and Hybrid systems with the help of manually crafted rules and machine learning techniques.

The first step of the procedure is the processing of text to change it into a standard form so as it becomes easy to access the token words.

There are several methods that accounts to the text processing where the unnecessary or meaningless words are eliminated. It also involves the creation of tokens. Tokens imparts meaning to the valuable statements and thus by referring to the tokens, one can easily understand the implication of the review. Although it must be done very carefully as sometimes it may lead to wrong conclusions. Therefore, one must be careful in the first and most important step of generation of tokens.

The next step includes classification of statements using different algorithms such as Decision Trees, Naive Bayes, Logistic Regression, Random Forest and k-nearest neighbour. The aim of classification is to predict the class of given data points. It utilizes the training data to understand how the input variables relate to that class. There are two types of learners- Lazy learners and Eager learners. The lazy learners store the training data and perform classification only when the testing data is loaded in the system. The Eager learners construct a model of classification based on the training data so that it becomes easy to classify the resting data once it appears.

Once the features are extracted, the text is fed to the model for training. It can be done by using the "Bag of Words" approach. This method is required to associate the features with values of different reviews. The Bag of Words converts the dataset into a matrix form where rows correspond to the reviews and columns refers to the tokens to be extracted from the review.

After the cells in the matrix is provided with the values, the quality of the machine learning algorithm is tested on the basis of:

978-1-7281-9393-9/20/$31.00 ©2020 IEEE
DOI: 10.1109/CICN.2020.06

- Accuracy
- F-measure
- Area under curve (AUC)
- Recall
- Precision

The major role of evaluation metrics is to determine the efficiency of the algorithm and to ensure that the model is working perfectly without any complications.

The paper is structured as follows: Section II describes the related work in the area of SA. Section III shows the proposed approach in a pointwise manner. Section IV includes the methods and techniques deployed. Section V presents the results of the proposed approach, and Section VI concludes the research and discusses future work.

II. RELATED WORK

The authors of [1] proposed to apply eight classifiers on the IMDb movie reviews dataset. These are - Naive Bayes, Decision Tree, Random Forest, Ripple Rule Learning, K-Nearest Neighbours, Support Vector Classifier, Bayes Net, Stochastic Gradient Descent. In their work Ripple Rule Learning was found to give the worst results, whereas Random Forest outperformed other classifiers. Performance of these eight classifiers were measured by five different evaluation metrics, namely, Accuracy, Area Under Curve (AUC), F1- measure, Recall, and Precision.

Authors of [2] were of the thought that reviews of movies shared on social media platforms and other web portals are important factors in a movie's financial success. The results showed that positive sentiment is more efficient for a movie domain with a small number of existing reviews, which indicated that sentiment alone is not the only factor. Rather, sentiment could perform better in combination with other factors such as movie genre and festive season etc.

In [3] the authors considered four classifiers; Maximum Entropy (ME), Naive Bayes (NB), Support Vector Machine (SVM), and Stochastic Gradient Descent (SGD) for SA on the IMDb dataset. Precision, recall, f-measure, and accuracy was used for performance comparison. Highlight of the work was that a longer "n-gram" got better results than a shorter one with all the classifiers.

Reference [4] focused on increasing performance of the Naive Bayes classifier. The authors combined negation handling, feature selection and word n-gram techniques to improve the performance. However their best result of 88.8% has been outperformed in the proposed approach.

In [5] a detailed study has been given on evaluation metrics. Use of multiple evaluation metrics is very crucial as an algorithm may perform well w.r.t a single evaluation method but may not give that good results when weighed against some other metric.

The authors of [6] aimed at performing SA on a multilingual system. For this purpose, they used lexical resources in the English SentiWordNet. For the work they had considered Amazon's German movie reviews dataset. Their approach got good results for a multilingual system.

Reference [7] unlike other citations, worked with pdfs, html files, xml files among others. It lists a number of methods for sentiment analysis in such systems.

In [8], a detailed study was made on how Conditional Random Field (CRF) and LR can be used for polarity check of collection of texts. They have deployed the proposed approach in the SemEval 2015 ABSA task.

Reference [9] contains a variety of ML approaches that can be considered for a SA task. The researchers thus provide a good and effective way to fix which algorithm to use in which case by describing different ways under a single work.

Indian languages are too used by a massive number of internet users. A study was made on such text material in [10]. The text was taken from Twitter and tweets were divided into positive, negative, and neutral ones.

III. PROPOSED APPROACH

The work gives a Sentiment Analysis model for classification of movies reviews taken from IMDb dataset. A pointwise summary of the proposed approach is given.

1. Retrieval of IMDb dataset from Kaggle Bag of Words Meets Bag of Popcorn Challenge.
2. Pre-processing of data including cleaning, HTML tags removal, stopwords removal etc.
3. Feature selection from the possible sets of features.
4. Text representation using BoW.
5. Fed to different classifiers.
6. Evaluated as per six different metrics.
7. Comparison done between all the classifiers deployed.

Following flowchart summarizes the approach being presented.

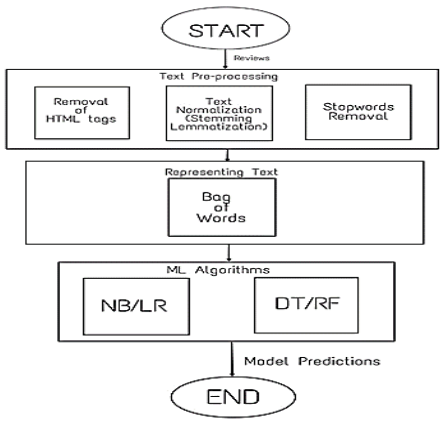

Fig. 1. Methodology Flowchart

IV. METHOD DESCRIPTION

The methods used in the work have been described in detail in this section.

A. Text pre-processing

Text gathered in the dataset is not ready to be fed to the classifier model as it is. This is primarily because classifiers need data in a prescribed form and not in the form of paragraphs. Also, the data has been received majorly from internet sources thus have a lot of html tags, abbreviations etc.

Following techniques were used to pre-process the data

- Removal of HTML tags - As stated before, the data has been primarily taken from the internet, it therefore consists some HTML programming part that needs to be scrapped before as they will not give any insights, rather if these go on to become vectors can pose a threat in terms of memory.

- Text Normalization - Movie reviews are generally a form of casual writing. For example, “good” at times may be written as “goooood”, “gud”, “gd” etc. All of these words will impart similar meaning but are available in different forms. These need to be converted in the base form called the canonical form. Additionally, a word may exist in different forms i.e. tenses and thus needs to be made fit for our purpose.

- Stopwords Removal - Common words such as “is”, “am”, “are”, “the” etc. are likely to give no meaning to the text. These words are used just to help the main meaning giving words. Such words are called stopwords. Natural Language Toolkit has a collection of such words from various languages. These can simply be removed.

B. Feature Extraction

- The concerned dataset has a huge amount of text, if taken in raw form it would turn out with an enormous number of features, thus feature extraction had to be used. Else, our ML model would have had overfitting issues.

- Feature extraction is alternatively called dimensionality reduction as we are limiting the number of dimensions that will represent our dataset. A feature can only be ignored in cases where they do not add any specific meaning to the data.

C. Text representation

After pre-processing and feature extraction are done the aim is now to transform the data in some way that could be understood by the classifier model. The models that were chosen required the text to be represented in some mathematical structure. Bag of Words (BoW) was picked up.

Bag Of Words

In BoW the complete dataset is converted into a matrix. This matrix is called DTM, Document Term Matrix. The rows in the matrix correspond to each review whereas the columns represent the words which form these reviews. Precisely, these words are rather n-grams. N-gram is a phrase having “n” words.

Values of the DTM cells can be filled in multiple ways, out of these two were explored

- Count - The cell is made to contain the actual count of occurrence of the word in the corresponding review.

- Term Frequency-Inverse Document Frequency(TF-IDF) - A statistical method of determining the importance of a word to a specific document (review in this case).

D. Classifiers

To classify the testing data into categories four classifiers were chosen

- Logistic Regression - Fits a hypothesis in the dataset which is a concrete form of mathematical sigmoidal function. It gives out the probability a review would fall in the positive category.

- Naive Bayes - Depends on conditional probability where the data to be labelled is thought of as a set of conditions that have occurred and the task is to tell the probability of a certain category.

- Decision Tree - A classifier model that gives labels to tokens based on a tree structure, where tree branches represent conditions on features, and tree leaves represent the label. [1]

- Random Forest - It is an extension of Decision Tree. Multiple decision trees are made with the root node being randomly selected. Condition is that the root nodes must have as little correlation as possible..

-

V. RESULTS

This section summarizes the results achieved.

TABLE I. RESULTS WITH COUNT

ML Algorithm / Metrics	**Features**				
	Accuracy	***Precision***	***Recall***	***F-score***	***AUC***
Logistic Regression	0.8728	0.8708	0.8777	0.8742	0.94
Naive Bayes	0.8594	0.8566	0.8658	0.8612	-
Decision Tree Classifier	0.7134	0.7218	0.7014	0.7114	0.71
Random Forest Classifier	0.8584	0.862	0.8558	0.8589	0.93

Table 1 depicts the results achieved with “count ” in the DTM as the cell value. Logistic Regression is the clear winner in AUC. Decision Tree does not work that well relative to other chosen classifiers.

TABLE II. RESULTS WITH TF-IDF

ML Algorithm / Metrics	Features				
	Accuracy	*Precision*	*Recall*	*F-score*	*AUC*
Logistic Regression	0.8914	0.882	0.9055	0.8936	0.96
Naive Bayes	0.8228	0.8285	0.8174	0.823	0.85
Decision Tree Classifier	0.7066	0.7098	0.7098	0.7114	0.71
Random Forest Classifier	0.8562	0.8597	0.8539	0.8568	0.93

Table 2 consists of the results after TF-IDF was chosen to fill DTM values. Logistic Regression again wins with a highly competitive AUC of 0.96.

The best results were thus achieved when TF-IDF values were filled in DTM and the classifier chosen was Logistic Regression. The graph below plots this case.

Fig. 2. AUC Curve for Logistic Regression

VI. CONCLUSION AND FUTURE WORK

In our work, we have presented an approach for the classification of the sentiments using the

IMDb dataset. Pre-processing of the dataset was done to make it suitable to be fed to the classifier model. Bag of Words approach was chosen for text representation in the work. Finally, four different traditional machine learning algorithms were deployed for getting results. These results were then compared on the basis of different evaluation metrics. Using

TF-IDF + Logistic Regression gave the best validation AUC of nearly 96% for our task.

In future, we can use state of the art word embeddings like Word2vec that overcomes some of the limitations of the TF-IDF based approach. Word2Vec captures the semantic similarity between the words and words with similar meanings are placed in close proximity in the vector space. However, it still suffers with one of the fundamental problems called Out of Vocabulary (OOV) which means that it is unable to provide an embedding for a word which is not in the training corpus. To deal with this we can use even more sophisticated word embeddings like Facebook's FastText or BERT etc. Also, to train these models we shall require more advanced hardware like GPUs and a higher degree of parallelization.

REFERENCES

[1] Yasen, Mais, and Sara Tedmori. "Movies Reviews sentiment analysis and classification." IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT), 2019.

[2] Mishne, Gilad and Natalie Glance, (2006), "Predicting Movie Sales from Blogger Sentiment", AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs.

[3] Abinash Tripathy, Ankit Agrawal, Santanu Kumar Rath, (2016), "Classification of sentiment reviews using n-gram machine learning approach", Expert Systems with Applications, Vol. 57, PP. 117-126.

[4] Vivek Narayanan, Ishan Arora, Arjun Bhatia, (2013), "Fast and Accurate Sentiment Classification Using an Enhanced Naive Bayes Model", Intelligent Data Engineering and Automated Learning – IDEAL, Springer, Vol. 8206.

[5] Hossin, Mohammad, and M. N. Sulaiman. "A review on evaluation metrics for data classification evaluations." International Journal of Data Mining & Knowledge Management Process 5.2 (2015): 1.

[6] Kerstin Denecke, (2008), "Using SentiWordNet for multilingual sentiment analysis", IEEE 24th International Conference on Data Engineering Workshop, Cancun, PP 507-512

[7] .Feldman, Ronen. "Techniques and applications for sentiment analysis." Communications of the ACM 56.4 (2013): 82-89.

[8] Hamdan, Hussam, Patrice Bellot, and Frederic Bechet. "Lsislif: Crf and logistic regression for opinion target extraction and sentiment polarity analysis." *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*. 2015.

[9] Sharma, Anuj, and Shubhamoy Dey. "A comparative study of feature selection and machine learning techniques for sentiment analysis." *Proceedings of the 2012 ACM research in applied computation symposium*. 2012.

[10] Prasad, Sudha Shanker, et al. "Sentiment classification: an approach for Indian language tweets using decision tree." *International Conference on Mining Intelligence and Knowledge Exploration*. Springer, Cham, 2015