ISSN: 0974-5823 Vol. 7 No. 6 June, 2022

International Journal of Mechanical Engineering

INTRIGUE TO ENCHANTING TEXT IN SENTIMENT ANALYSIS

Preeti Arora[1*], Dr. B.D.K.Patro[1]

[1*] Research Scholar, [1] Associate Professor

Maharishi University of Information Technology, Lucknow.

ABSTRACT

Due to the large diversity and amount of social media data, it is difficult to identify the newest trends and summarize the state or general opinions on products, necessitating the use of automated and real-time opinion extraction and mining. Online opinion mining is a type of sentiment analysis that is approached as a challenging text classification job. We investigate the significance of text pre-processing in sentiment analysis, and offer experimental findings that show that with proper feature selection and representation, sentiment analysis accuracy using support vector machines (SVM) may be greatly improved. Although sentiment analysis is considered to be a considerably more difficult task in the literature, the degree of accuracy reached is equivalent to that achieved in subject classification.

Keywords: Sentiment Analysis; Text Pre-processing; data; Big Data

I. INTRODUCTION

Big data is one of the most important topics in the world right now. There has been a significant surge in social media giants such as Twitter in recent years, revealing them to be vast amounts of big data. These data may then be gathered in massive quantities and used to train machine learning and Deep Learning algorithms that will help in decision-making.

Sentiment analysis is a technique for extracting text from a variety of sources for personal or business purposes. Because of social media's ubiquity, everyone publishes a vast amount of data online, which may subsequently be utilized to produce feelings. This may be used to provide firms with information on their products. Information like as product performance throughout the course of the year, competitor analysis, and so on may be retrieved and used to the company's benefit.

Phan et al. investigated the use of tweets to identify real-time drug misuse. The authors used a dataset of legal and illicit medications, as well as actual content from 31,478 tweets. It employs the J48, Random Forest, Nave Bayes, and SVM (Support Vector Machine) classifiers for training and does not need any preprocessing. The J48 method was used to test the constructed classifier on a real-world twitter dataset, which yielded a precision of 74.8 percent. The proposed study involves the use of Phrase Frequency-Inverse Document Frequency (TFIDF) to reflect the significance of a term in a given document and increase accuracy, as well as the use of Mechanical Turk for large-scale data collecting.

The act of researching internet reviews in order to determine the general feeling or opinion about a product is known as sentiment "analysis of reviews." Owing to the diversity and scale of social media data, it is difficult to find out the newest trends and summarize general opinions due to the enormous number of reviews on the internet. This necessitates the need for real-time opinion extraction and mining. Because of the subjective nature of evaluations, deciding on an emotion is a difficult task.

II. PROCESS OF SENTIMENT ANALYSIS

Figure 1 depicts the process of "sentiment analysis," which includes gathering customer evaluations and preprocessing data, then aspect identification, sentiment classification, aspect ranking, and overall sentiment rating.

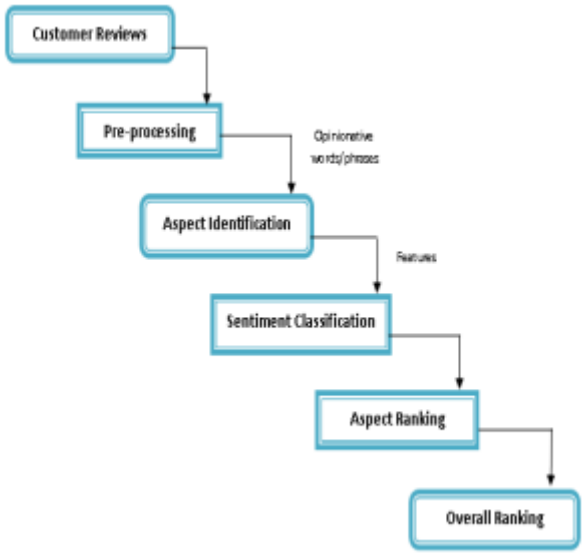

Figure 1: Sentiment analysis process

Each stage in Figure 1 "displays data in a different condition; the first state displays the raw data, which may contain incorrect data types or labels, special characters, white spaces, and varying font sizes. Without any pre-processing, this data cannot be immediately put into the model for analysis. The data becomes technically valid after this pre-processing, which means it can now be fed into the model without problems.

"Pre-processing" means "filling in missing values, smoothing noisy data, eliminating outliers, and resolving discrepancies in data acquired from primary or secondary sources for analysis or model construction."

The process of cleaning and preparing the text for categorization is known as pre-processing. This phase is required since online texts frequently contain noise and non-informative elements such as HTML tags, scripts, and ads. Furthermore, many words in the text have little effect on the overall orientation of the text at the word level. Because each word in the text is viewed as a separate dimension, preserving irrelevant terms increases the problem's dimensionality, making classification more complex. These issues show themselves not just in the analysis's robustness, but also in the classification process's computing complexity. Text preparation is not only a necessary aspect of the data science process, but it also takes up the most time.

III. FRAMEWORK

We propose a computational framework for sentiment analysis that is divided into three steps. First, by utilising significant data processing and filtering, the most relevant attributes will be retrieved. Second, on each of the feature matrices generated in the first phase, classifiers will be developed using SVM, and the accuracies resulting from the prediction will be obtained, and finally, the classifier's performance will be compared to other ways.

The most difficult aspect of the framework is feature selection, which we go through in detail here. We'll start by cleaning up the HTML elements, expanding abbreviations, removing stopwords, managing negations, and stemming the data, all of which will be done using natural language processing techniques. Based on three distinct feature weighting algorithms, three separate feature matrices are produced (FF, TF-IDF and FP). We then proceed to the filtering step, where we compute the chi-squared statistics for each feature inside each document and pick relevant features using a specific criterion, followed by the generation of other feature matrices using the same weighting procedures as before.

The data consists of two movie review data sets: one comprising 1400 documents (700 positive and 700 negative) (Dat-1400), and the other including 2000 documents (1000 positive and 1000 negative) (Dat-2000) (Dat-2000). Both sets are open to the public. Despite the fact that the first set is contained in the second, they were analyzed individually because to the diverse collection of factors that might impact the text. This separation also allows for a fair comparison with research that employed them individually. Unigrams are the feature type employed in this investigation. The data is processed in the following manner.

3.1 Data Transformation

Any HTML tags had previously been removed from the text. Pattern recognition and regular expression algorithms were used to extend the abbreviations, and the text was subsequently cleansed of non-alphabetic signals. When it came to stop words, we compiled a list from several existing standards stop lists, with a few tweaks to account for the data's unique properties. In movie reviews, the terms film, movie, actor, actress, and scene, for example, are non-informative. Because they are movie domain particular terms, they were termed stop words.

In the case of negation, we began by tagging the negation word with the subsequent words until we reached the first punctuation mark. In the classifier, this tag was employed as a unigram. There was not much of a change in the findings when comparing the results before and after adding the labelled negation to the classifier. This conclusion is supported by the evidence. The reason for this is that finding a match between the labelled negation words across all of the papers is difficult. As a result, we decreased the tagged words after the negative to three, then to two words, taking syntactic position into consideration, allowing additional negation phrases to be included as unigrams in the final collection of reduced features.

In order to remove repetition, the papers were also stemmed. The number of features was lowered from 10450 to 7614 in Dat-1400, then from 12860 to 9058 features in Dat-2000. Following that, for each dataset, three feature matrices were created using three distinct methods of feature weighting: TF-IDF, FF, and FP. To be explicit, the (i, j)-th element in the FF matrix represents the FF weight of feature i in document j. Experiments were conducted on the Dat-1400 feature matrices, which will be displayed in Section 4.

3.2 Filtering

The filtering approach we're utilizing is the univariate chi-squared method. It is a statistical analytic approach used in text classification to determine the relationship between a word and the document category in which it appears. The chi-squared score is low if the term appears often in several categories, but high if it appears frequently in a few categories.

The chi-squared test value was obtained for each characteristic of the first stage's produced features in this stage. Following that, a final set of features was chosen in both datasets based on a 95 percent significance level of the value of chi-squared statistics, resulting in 776 out of 7614 features in Dat-1400 and 1222 out of 9058 features in Dat-2000. The features matrices on which the classification was done were built using the two sets. Each data set has three feature matrices at this point: FF, TF-IDF, and FP.

3.3 Classification Process

We use the SVM classifier on each stage after creating the matrices indicated above. The Gaussian radial basis kennel function was chosen because it offers a parameter that regulates the area in the data space where the support vector has an influence. The machine learning package 'e1071' in R was used to apply SVM. Due to the sensitivity of SVM performance to their values, we used it with numerous combinations of C and. Each set was separated into two sections for the classification process, one for training and the other for testing, in a 4:1 ratio, meaning 4/5 portions were utilised for training and 1/5 for testing. Then, for classification, training was done using 10 folds cross validation.

3.4. Performance Evaluation

Precision, recall, and F-measure are the performance measures used to assess the categorization outcomes. True positive (tp), false positive (fp), true negative (tn), and false negative (fn) assigned classes are used to calculate these measures. Precision is defined as the number of true positives out of all positively assigned documents divided by the total number of documents.

$$precision = \frac{tp}{tp+fp} \quad (1)$$

The number of true positives out of the total number of positive documents is known as recall.

$$recall = \frac{tp}{tp+fn} \quad (2)$$

Finally, the F-measure is a weighted accuracy and recall approach that is calculated as

$$F - measure = \frac{2+precision+recall}{precision+recall} \quad (3)$$

where the number varies from 0 to 1 and the closer it gets to 1 the better the outcomes.

IV. EXPERIMENTS AND RESULTS

In this part, we present the results of many experiments used to evaluate the classifier's performance. We evaluate the performance of the classifier on each of the features matrices resulting from each data transformation and filtering to that of the classifier on non-processed data using accuracies and Equation 1. We also compare the findings to the published results in terms of accuracies and feature types.

"Standard machine learning classification approaches, such as support vector machines (SVMs), may be used to the full documents themselves," according to the argument, which is why the classifier is applied to the entire texts without any preprocessing or feature selection methods. As a consequence, we categorized the documents without any pre-processing to allow a fair comparison with previous findings based on the adjusted kernel parameters we are employing in this stage, $\gamma = 0.001$ and $C = 10$. The classifier was then applied to the Dat-1400 characteristics matrix obtained during the first step of pre-processing.

For each of the features matrices, Table 1 compares the classifier results resulting from classification on both unprocessed and preprocessed data (TF-IDF, FF, FP). It also compares these findings to those obtained using both the TF-IDF and FF matrices. The comparison is made using the achieved accuracies as well as the metrics derived in Equations 1, 2, and 3.

Table 1: The classification accuracies in percentages on Dat-1400, the column no pre-proc refers to the results reported, no pre-proc2 refers to our results with no pre-processing, and pre-proc refers to the results after pre-processing, with optimal parameters $\gamma=10^{-3}$, and C=10

	TF-IDF		**FF**			**FP**		
	no pre-proc	**pre-proc**	**no preproc1**	**no preproc2**	**pre-proc**	**no preproc1**	**no preproc2**	**pre-proc**
Accuracy	78.33	81.5	72.7	76.33	83	82.7	82.33	83
Precision	76.66	83	NA	77.33	80	NA	80	82
Recall	79.31	80.58	NA	76.31	85.86	NA	83.9	83.67
F-Measure	77.96	81.77	NA	76.82	82.83	NA	81.9	82.82

Table 1 reveals that for data that was not subjected to pre-processing, the FF matrix accuracies improved significantly, from 72.8 percent recorded in to 76.33 percent, but the FP matrix accuracies were marginally different, with 82.33 percent attained vs 82.7 percent reported. In addition, where [3] did not employ TF-IDF, we got 78.33 percent accuracy in the TF-IDF matrix. By digging further into the data, we can see that when

the classifier is applied to the pre-processed data following the data transformation, the accuracy increases, with the greatest accuracy of 83 percent for both matrices FF and FP.

Table 1 shows that, while the accuracy achieved in the FP matrix is similar to that achieved previously and in, there is a significant difference in the classifier performance on the TF-IDF and FF matrices, demonstrating the importance of stemming and removing stop words in improving sentiment classification accuracy. We emphasize the need of designing and using a kernel for that specific challenge in order to employ the SVM classifier across the full document.

The three separate matrices that were produced following the filtering are then classified (chi-squared feature selection). In comparison to what was obtained in prior experiments and in, the classifier's successes (see Table 2) were impressive. By selecting features based on their chi squared statistics value, we were able to reduce the dimensionality and noise in the text, resulting in a high classifier performance that was equivalent to topic classification.

Table 2 shows the classifier performance accuracies and assessment metrics before and after chi squared was applied.

Table 2: The classification accuracies in percentages before and after using chi-squared on Dat-1400, with optimal parameters $\gamma=10^{-5}$, and C=10

	TF-IDF		**FF**		**FP**	
	no chi	**chi**	**no chi**	**Chi**	**no chi**	**Chi**
Accuracy	81.5	92.3	83	90	83	93
Precision	83	93.3	80	92	82	94
Recall	80.58	91.5	85.86	88.5	83.67	92.16
F-Measure	81.77	92.4	82.83	90.2	82.82	93.06

Table 2 demonstrates a considerable improvement in classification quality, with the FP matrix achieving the greatest accuracy of 93 percent, followed by 92.3 percent in TF-IDF and 90 percent in FF matrices, with the F-measure values being extremely near to 1, indicating good classification performance. To the best of our knowledge, those results have never been published in previous research utilizing chi-squared sentiment analysis at the document level. As a result, using transformation and subsequently filtering on text data minimizes noise in the texts and improves classification performance. Figure 1 demonstrates how the accuracy of SVM prediction improves as the number of characteristics decreases.

A feature relation networks selection based technique (FRN) for selecting related features from Dat-2000 and improving sentiment prediction using SVM. The accuracy attained with FRN was 89.65 percent, compared to 85.5 percent using the chi-squared approach, as well as other univariate and multivariate feature selection methods.

We pre-processed Dat-2000 before running the SVM classifier, resulting in a high accuracy of 93.5 percent in the TF-IDF matrix, 93 percent in FP, and 90.5 percent in FF (see Table 3), which is greater than what was discovered in.

Table 3: Best accuracies in percentages resulted from using chi-squared on 2000 documents. with optimal parameters $\gamma=10^{-6}$, and C=10

	TF-IDF	**FF**	**FP**
Accuracy	93.5	90.5	93
Precision	94	89.5	91
Recall	93.06	91.3	94.79
F-Measure	93.53	90.4	92.87

The characteristics employed are of many sorts, including several N-gram categories such as words, POS tags, Legomena, and so on, whereas we simply use unigrams. We have shown that utilising unigrams in classification has a greater influence on classification outcomes than using other feature types, which is in line with the findings.

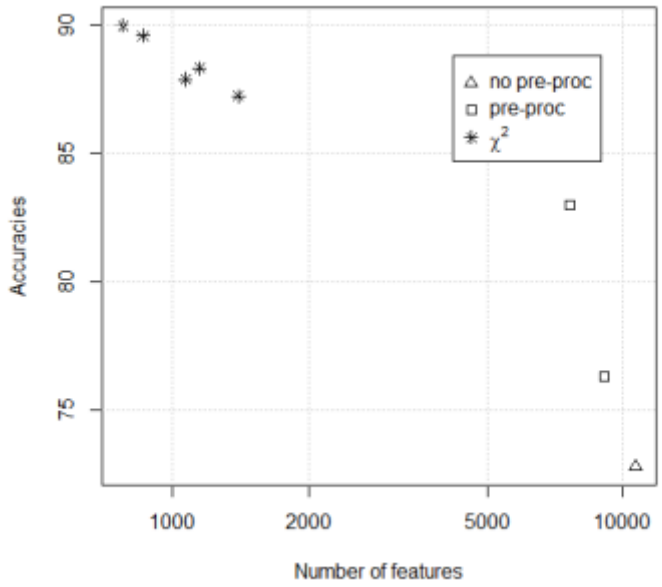

Figure 1: The correlation between accuracies and the number of features, no pre-proc refers to the results in, pre-proc and χ2 refers to our results

V. CONCLUSION

Sentiment analysis appears to be a difficult area with many challenges because it requires natural language processing. Its findings may be used in a wide range of applications, including news analytics, marketing, question answering, knowledge bases, and so on. The goal of this discipline is to improve the machine's capacity to comprehend texts in the same way as human readers do. For many firms and institutions, getting valuable insights from viewpoints published on the internet, particularly through social media blogs, is critical, whether it's in terms of product feedback, public mood, or investor opinions.

We looked at the emotion of internet movie reviews in this paper. To decrease the noise in the text, we employed a variety of various pre-processing approaches, as well as the chi-squared method to eliminate extraneous characteristics that did not impact the text's orientation. Extensive experimental findings suggest that proper text pre-processing approaches, such as data transformation and filtering, may improve the classifier's performance greatly. The amount of accuracy attained on the two data sets is equivalent to that which can be reached in subject categorization, which is a considerably simpler task.

REFERENCES: -

1. Choudhary Nidhi.2014. A Study over Problems and Approaches of Data Cleansing/Cleaning,. Volume 4, Issue 2, February 2014 .
2. Chen, Z., Mukherjee, A., Liu, B., Hsu, M., Castellanos, M., and Ghosh, R.2013. Exploiting domain knowledge in aspect extraction. In EMNLP, pages 1655–1667. 2013.
3. Ibrahim Housien Hamed, Zuping Zhang & Qays Abdulhadi Zainab.2013. A comparison study Of Data Scrubbing algorithm and framework in Data Warehousing. International Journal of Computer Applications (0975 – 8887) April 2013.
4. Y.Patil Rajashree, Dr. Kulkarni R.V.2012. A Review of Data Cleaning Algorithms for Data Warehouse Systems. IJCSIT , Vol. 3 (5) , 2012.

5. M. Thelwall, K. Buckley, G. Paltoglou, Sentiment in twitter events, Journal of the American Society for Information Science and Technology 62 (2) (2011) 406 418.
6. A. Abbasi, S. France, Z. Zhang, H. Chen, selecting attributes for sentiment classification using feature relation networks, Knowledge and Data Engineering, IEEE Transactions on 23 (3) (2011) 447 462.
7. L. Tan, J. Na, Y. Theng, K. Chang, Sentence-level sentiment polarity classification using a linguistic approach, Digital Libraries: For Cultural Heritage, Knowledge Dissemination, and Future Creation (2011) 77 87.
8. Tang, H., Tan, S., Chang, X. 2009. A Survey on sentiment detection of reviews. Expert Systems with Applications 36(7) (2009) 10760-10773
9. P. Melville, W. Gryc, R. Lawrence, Sentiment analysis of blogs by combining lexical knowledge with text classification, in: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2009, pp. 1275 1284.