Comparative Analysis of Lexicon and Machine Learning Approach for Sentiment Analysis

Roopam Srivastava[1]
Computer Science Dept.
Shri Venkateshwara University
Gajraula (U.P.), India

Prof. (Dr.) P.K. Bharti[2]
Computer Science Dept.
Shri Venkateshwara University
Gajraula (U.P.), India

Dr. Parul Verma[3]
Amity Institute of Information Technology, Amity University
Lucknow, Lucknow (U.P.), India

***Abstract*—Opinion mining or analysis of text are other terms for sentiment analysis. The fundamental objective is to extract meaningful information and data from unstructured text using natural language processing, statistical, and linguistics methodologies. This further is used for deriving qualitative and quantitative results on the scale of 'positive', 'neutral', or 'negative to get the overall sentiment analysis. In this research, we worked with both approaches, machine learning, and an unsupervised lexicon-based algorithm for sentiment calculation and model performance. Stochastic gradient descent (SGD) is utilized in this work for optimization for support vector machine (SVM) and logistic regression. AFINN and Vader lexicon are used for the lexicon model. Both the feature TF-IDF and bag of a word are used for classification. This dataset includes "Trip advisor hotel reviews". There are around 20k reviews in the dataset. Cleaned and preprocessed data were used in our work. We conducted some training and assessment. A classifier's accuracy is measured using evaluation metrics. In TF-IDF, the Support Vector Machine is the more accurate of the two classifiers used to assess machine learning accuracy. The classification rate in Bag of Words was 95.2 percent and the accuracy in TF-IDF was 96.3 percent on the support vector machine algorithm. VADER outperforms the Lexicon model with an accuracy of 88.7%, whereas AFINN Lexicon has an accuracy of 86.0%. When comparing the Supervised and unsupervised lexicon approaches, support vector machine model outperforms with a TFIDF accuracy of 96.3 percent and a VADER lexicon accuracy of 88.7%.**

Keywords*—*NLP; sentiment analysis; SGD (stochastic gradient descent); machine learning; TFIDF; BoW; VADER; SVM; AFINN

I. Introduction

With the use of natural language processing, data from online reviews can be leveraged to extract business intelligence. It is an area of artificial intelligence and linguistics-focused on teaching computers to understand human language statements or words. It was designed to make users' life easier and also help them in communicating easily with the system using natural language [1]. The application of sentiment analysis is very broad and powerful, such as Expedia Canada; Canadians employ sentiment analysis when they see that people complain about their television station's music. Rather than dismissing a bad comment, Expedia capitalizes on it by airing all-new soulful music on their channel [2]. Supervised and unsupervised learning are two machine learning methodologies for sentiment classifiers. In a supervised technique, the classifier requires labeled training data as well as the target. In the present study, sentiment classification is done using a supervised and unsupervised approach. Calculating the values of parameters of functions that minimize a cost function using 'stochastic gradient descent is a simple yet effective optimization strategy that converges on a solution to a problem by selecting an arbitrary solution, examining the goodness of fit (under a loss function), and then stepping in the direction that minimizes loss. Support vector machine and logistic regression classifiers for this model's accuracy are used in the first approach. Support vectors are the coordinates of each unique observation, to put it simply. The SVM classifier is a frontier that effectively separates the two classes (hyper-plane/line). The decision function only uses a subset of training points, making it memory efficient. Logistic regression is a supervised learning algorithm used most commonly to solve binary classification problems. The model might be developed using supervised learning to read the data and predict sentiment. More specifically, classification models would be used to solve the challenge. In another approach, using unsupervised lexicon-based models like 'afinn lexicon', Vader lexicon is used for sentiment analysis lexical model is a vocabulary of words that have been specifically matched for sentiment analysis, frequently including positive and negative phrases, as well as the magnitude of the polarity. We used the TF-IDF and BoW for feature engineering. Word embedding is a vector-based technique that represents text as a vector. To evaluate the classification system's accuracy, different evaluation measures, such as the F-Score and Accuracy score, have been employed. For text normalization, we employed various preprocessing steps like tokenization, stop words, lemmatization, n-gram, and punctuation removal to increase our system's performance. In this paper, Section II explains the related work and Section III tells the method, and workflow diagram in Section IV describes the result and discussion. Section V is the conclusion.

II. Literature Review

One can understand the analysis of sentiment as a type of data mining using computational linguistics, NLP, and text analysis for examining people's feelings. There are primarily two methods for extracting sentiment from reviews and categorizing the outcome as good or negative. Machine learning and lexicon-based approaches are examples. The lexicon-based strategy necessitates a predetermined lexicon, but the Machine Learning approach automatically classifies the review and thus needs training data. Here, a task related to it is discussed. Utilizing an existing generated annotated corpus, using citation sentences, this study analyzes the sentiment expressed in scientific articles. There are 8736 citation sentences in this corpus. They used the classification method to create six different machine learning algorithms. The system's accuracy is then assessed using various evaluation indicators. Using n-gram features in SVM classifier, the author showed commendable accuracy with micro-F. In comparison to the baseline system, their solution enhanced performance by a maximum of nine percent [2]. This paper provides a framework for automatically classifying internet news articles and reviews several existing approaches for classifying online news articles. Various classifiers were tested to get high accuracy. Using a Bayesian classifier, the experimental technique obtained the best accuracy in terms of confusion measures [3]. It is an automated text classification that has long been seen as an essential tool for organizing and analyzing massive volumes of digital documents that are widely dispersed and expanding. It has been discovered that the classification performance of classifiers based on different training text corpus differs, even for the same classification strategy, and that these differences might be quite considerable in some cases [4]. In this study use of an imbalanced and multi-classed data set of large size was made to determine an effective approach for sentiment analysis. Both features, bag-of-words, and tf-idf together with multiple machine learning algorithms (SVM, LR, MultinomialNB, Forest Tree) were used. Using support vector machine and logistic regression with BoW techniques, their best approaches outperform well on SVM and LR [5]. To classify movie reviews, this article employs NLTK, Text Blob, and the VADER Sentiment Analysis Tool. The results of this study's experiments show that Vader outperforms in comparison to text blob [6]. They show how to extract sentiment from text using a lexicon-based technique. The Semantic Orientation CALculator (SO-CAL) integrates intensification and negation and uses dictionaries of words tagged with their semantic orientation (polarity and strength). SO-CAL is used in the polarity classification task, which entails labeling a text with a positive or negative label that reflects the text's attitude toward its major subject matter. It demonstrates that SO-performance CAL's is consistent across domains and in data that has never been seen before.[7] Researchers devised a multi-classification technique for studying tweets, and they used Vader to categorize tweets on the 2016 US election. According to the results, this Sentiment Analyzer was a good choice using Twitter data for sentiment analysis classification. A large amount of data could be classified rapidly by using VADER [8]. The use of a Rule-based classification system for improving sentiment analysis in online communities is also feasible. In addition to general-purpose sentiment analysis, researchers employ emoticons, modifiers, SWN-based sentiment classification, and domain-specific phrases to analyze evaluations within online communities. A disadvantage of this strategy in terms of classification efficacy for domain-specific words is the need for automatic classification and scoring of words [9]. In this study, the next word negation is used to classify the sentiment of text using frequency-inverse document frequency. For text classification, the binary model of a “bag of words, tf-idf, and TF-IDF-NWN model” was also compared [10]. To automatically evaluate sentiment polarity and score, this method used an upgraded bag-of-words model that used word weight instead of term frequency to evaluate sentiment polarity and score. This technique may also classify reviews based on scientific topic area traits and keywords. This provided solutions to typical sentiment analysis issues that are suitable for use in a review system [11]. LeSSA was a new framework for textual sentiment classification that they had created. He made three key contributions: he established the K-means cluster from lexicon creation, offering a high-quality, broad-coverage sentiment lexicon, and he employed three strategies to build a high-quality training dataset for classification models. In terms of classification accuracy, their approach exceeds previous semi-supervised learning strategies [12]. In their research, they used four classifiers for sentiment analysis optimization: naive bayes, ‘OneR’, ‘BFTree’, and ‘J48’. In terms of precision, F-measure, and correctly classified cases, OneR appears to be more promising than others. [13]. They used the word embedding technique is word2vec in their model for the word vector. Then applied the LDA model with weighted tf-idf. Their approach showed b [14].

III. Methodology

This section establishes the methodology's goal. Fig. 1 illustrates our process. We used the “trip advisor hotel review” dataset in my work. One can examine what constitutes a wonderful hotel with this dataset, which has 20k reviews scraped from Trip Advisor which was downloaded from kaggle.com. It has two columns ‘Review' and 'Rating'. Five ratings appear in the rating column. Positive reviews receive a rating of (4,5), negative reviews (1,2), and neutral reviews (3) [15]. Our analysis only considers positive and negative reviews in our dataset. It is a comma-separated (.csv) file. We utilized the ScikitLearn python machine learning library, and for text processing, NLTK library from natural language processing for implementing the system. First, we do, data Pre-processing, calculating sentiment, features, and classification are all part of the classification process. In the unsupervised method, we used preprocessed data, then extract the data, model generation, calculating polarity score, and predicted sentiment.

Data Pre-Processing:

It is one of the initial steps in the feature engineering and modeling process. During the pre-process we clean the data, and normalize the corpus which has phrases and words into a standard form. This allows for document corpus standardization, which aids in the development of critical features and noise reduction caused by unwanted objects. We utilized the NLTK tool kit to perform data preprocessing. We go through the following procedures during test preparation, which is listed below:

- Cleaning Text-Unnecessary content, such as HTML tags, frequently appears in our text, adding little value to sentiment analysis. As a result, we must ensure that they are removed before extracting features.

- Lower Case-Because the computer sees lower case and upper case differently, if the text is in the same case, the machine can simply comprehend the words. To avoid problems like these, we should make all of the text in the same case, with a lower case being the best option.
- Remove special characters and digits-This is another text preprocessing strategy that can handle the words 'hurray' and 'hurray!' or game45. Because this type of word is difficult to digest, it is preferable to eliminate it or replace it with an empty string. For this, we employ regular expressions.
- Tokenization - Converting sentences into words.
- Stopword Removal - Stopwords are the most common words that provide no meaningful information in a text. It includes words like 'they', 'there', 'this', 'there',' a', 'an', and 'the'. NLTK library is a commonly used library for stopword removal. We can quickly add any new word to a list of terms by using the added technique. The function removes stopwords () helps eliminate stopwords from a corpus while keeping the most important and contextual words.
- Lemmatization - In the same way, as stemming removes affixes words to get to a word's fundamental form, lemmatization does the same. In actuality, it's a technique for reducing words to their lemma by comparing them to a linguistic dictionary. WordNetLemmatizer is a tool provided by nltk. The stem is commonly used for lemmatization. Now we get the clean review for further procedure.

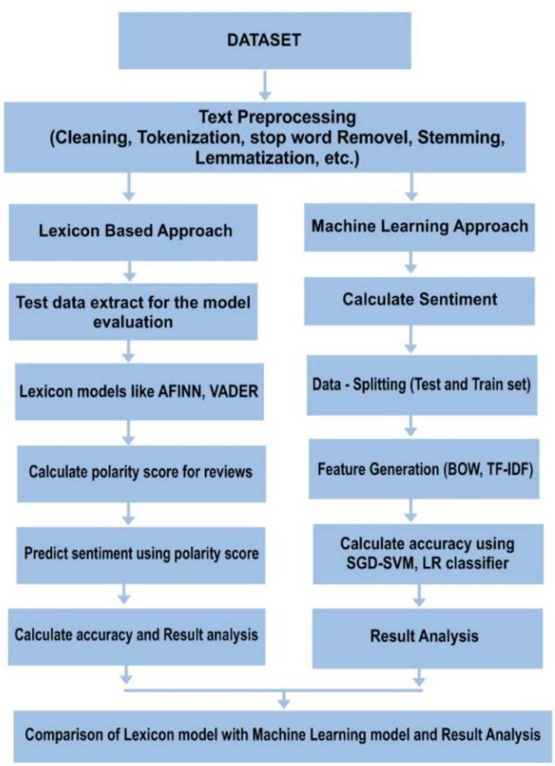

Fig. 1. The Flow of System Work.

A. *Sentiment Calculation for ML Model*

We calculate sentiment over 'Rating column in the dataset during supervised learning. Based on the rating column, we estimated sentiment. '1' denotes a positive sentiment, whereas '0' denotes a negative sentiment, the result of sentiment as depicted in Table I.

TABLE I. SENTIMENT EXAMPLE

Reviews	**Sentiment**
unique, great stay, wonderful time hotel monaco...	1(positive)
ok, nothing special charge diamond member hill...	0(negative)

B. *FeatureEngineering for Supervised Machine Learning Models with Bag of Word and TF-IDF*

The process of transforming raw data into attributes helps aid predictive models in gaining a deeper understanding of the situation, resulting in enhancing the accuracy of previously unknown data. This is also known as feature engineering. The goal of feature-selection approaches is to reduce the dataset's dimensionality by deleting features that aren't essential to the classification [16]. A bag- of- word is converted text into vectors using the count vectorizer function. BOW extracts words from a text and creates a list of all the words and their frequency. To put it another way, a dictionary of all the words in the text is constructed. Because the structure of words and their meaning in context is gone, it is referred to as a bag of words. The combination of sequenced words in a text is referred to as an n-gram, with n denoting the number of words in the combination. When N equals 1, shows the text has a single word. If N equals 2, it refers to a pair of words that have been sequenced. In our classification, we used many types of N-grams, each of which yielded different results. Table II shows N-grams with various N values as an example based on the sentence "I like to eat pizza". The feature is employed as a bigram in our model.

TABLE II. N-GRAM

Value of N	**Gram-Value**	**Example**
N equals 1	Uni-gram	I, like, to, eat, pizza
N equals 2	Bi-gram	I like, like to, to eat, eat pizza
N equals 3	Tri-gram	I like to, like to eat, to eat pizza

1) Term Frequency-Inverse Document Frequency (TF-IDF) -The primary premise behind that words that occur more frequently in a document are given more weight than terms that appear less frequently. The frequency of each term is referred to as term frequency in this case. The tf-idf model performs effectively and prioritizes rare words over the binary bag of words approach, which treats all words equally [10]. Term frequency displays the significance of the word to a document, based on the assumption that the more terms in the document, the greater the importance.

$$tf = frequency\ of\ a\ word/total\ word$$

Inverse document frequency demonstrates how a term is genuinely useful. It is not required that a phrase that appears frequently in some documents, such as stopwords, be relevant (the that, of, etc.). Stopwords obscure the context and should therefore be avoided. IDF operates in such a way that they are completely ignored calculated by:

$$IDF = \ln(total\ no.of doc\ /no.of\ doc.that\ contain\ term)$$

2) Calculate TF-IDF for matrix generation- The tf-idf score (w) for a word in a corpus document is obtained by combining these two features. To create a composite weight for every phrase in each document by using the tf-idf model. Term 't' is given a weight in the document 'd' via the tf-idft weighting technique. When 't' appears repeatedly in a small number of documents, it has the maximum impact.

$$(tf - idf\ )_{t,d} = tf_{t,d} \times idf_t$$

When a term appears fewer or more times in a document, it is considered lower.

C. Classification Classifiers

The next stage is to use classification algorithms after preprocessing and feature selection. In the literature, several text classifiers have been proposed [17]. We employed machine learning algorithms such as SGD-Support Vector Machine (SVM) and Logistic Regression (LR).

- Support Vector Machine - Creates a decision boundary that is as robust as possible by using linearly separable classes. This indicates that the position of the boundary is determined by the points nearest to it. The decision boundary is a line or hyperplane that is as far away from either class's nearest training instance as possible. The SVM algorithm is a constraint-based optimization problem with inequality constraints. To address this problem, we employed support vector machine optimization with a hard margin (SGD).
- Stochastic Gradient Descent - Updates a set of coefficients by taking a "step" of a certain size in the opposite direction to the gradient, determining the gradient of the loss function at a specific point in the dataset, and updating the coefficients. The method modifies the coefficients iteratively, moving them away from the steepest ascent and toward the minimum, emulating a solution to the optimization issue.
- LR (Logistic Regression)- It is used when the dependent variable (target) is categorical. For binary and linear classification challenges, it is a simple and effective strategy. It's a straightforward classification model that produces outstanding results with linearly separable classes [18].

On a training review, build BOW and TF-IDF features, then transform test reviews into features and get the train and test shape. Using Logistic Regression and the SGD classifier for both features, before testing the model's performance, we fitted it to the train set and used predict to make predictions.

D. Sentiment Analysis using Unsupervised Lexicon-Based Models

This methodology stores specific information about words and phrases, such as sentiment polarity, objectivity, and subjectivity, with well knowledge bases, ontologies, lexicons, and databases. Many sentiment analysis methods rely heavily on an underlying opinion. "Lexicon features lists that are generally labeled according to their semantic orientation as either positive or negative is called sentiment lexicon" [21]. These lexicons frequently incorporate both positive and negative scores. There are a variety of popular lexical models for sentiment analysis. Some examples include the afinn and the Vader.

- Afinn Lexicon- It is one of the most basic and frequently used for sentiment analysis. It contains about 3300 words, each of which has a polarity score. The greatest features for conducting Twitter Sentiment Analysis are AFINN and Senti-strength. As a result, they're an excellent starting point for Twitter Sentiment Analysis [19].
- VADER Sentiment Lexicon- The sentiment dictionary with Valence The human-validated reasoning sentiment lexicon is of gold-standard quality [20]. It is open-source and included in the NLTK package, allowing it to be used directly on unlabeled text data. It is capable of detecting emotional polarity and intensity. It's a sentiment analysis model that can analyze a text by considering the text emotion's positive/negative polarity and its intensity. A decimal (float) value in the range [-1,1] indicates the text's polarity. It expresses the sentence's positive tone. When the polarity is less than zero it denotes negative polarity otherwise positive.

E. Sentiment Evaluation using Lexicon Model

Our unsupervised model, we used AFINN and VADER lexicon. We must first clean our data before proceeding with our analysis. It refers to the process of pre-processing and normalizing the text for analysis, which we have done earlier. Tokenized sentences are matched with words in the model to determine context and sentiment if any. We use a combining function such as sum or average to determine the final prediction about the overall text composition. In our work, using preprocessed data, we extract test reviews and test sentiments data for model evaluation. We then apply the above lexicon models to the reviews and calculate the polarity score as shown in Tables IIIA and IIIB.

TABLE III. (A). SENTIMENT AND POLARITY SCORE USING AFINN

Sample-Review	**Sentiment**	**score**
shame hotel wasn'tgood restaurant, arrived......	negative	-0.5
great location, partial ocean view room larger…….	positive	19.0

(B).SENTIMENT AND POLARITY SCORE USING VADER

Sample-Review	Sentiment	score
shame hotel wasn't good restaurant, arrived......	negative	-0.3
great location, partial ocean view room larger…….	positive	28.9

Using the polarity score, we predicted sentiment for review data, evaluated model performance, and predicted sentiment for positive and negative classes.

IV. RESULT AND DISCUSSION

We have done so to complete the experimental task. Due to any process, the maximum dataset which is to be generated is imbalanced. Using the above-mentioned data set, there are two columns in the dataset that is 'Review' and 'Rating'. Taking positive and negative reviews which have ratings (4,5) for positive and (1,2) for negative from the dataset. We used an imbalanced dataset with two classes ('positive' and 'negative') for work. Ten thousand sample reviews are taken from the dataset for a model. Review is preprocessed using the NLTK tool. The sentiment is calculated over the 'Rating label'. Training and testing sections of the dataset are separated, with test data making up 30% of the total for both machine learning and lexicon model, but only test data is used in the lexicon model. Various algorithms of machine learning are used for classification. The sentiment of the target dataset is utilized to generate features. In ML, feature generation was done by Bow and tf-idf, model is generated using classifiers such as SGD-SVM and logistic regression for accuracy calculation Stochastic Gradient Descent is used to solve hard margin support vector machine optimization. A fit function is used in the train set to fit the model and the prediction function applies to the test set, and objects were created for these functions. Can see in the Table IV(a) applied SVM and logistic regression as classifiers using a bag of word features for accuracy calculation. Table IV(b) shows that TFIDF features are used by both classifiers (SVM and LR) for accuracy. Stochastic Gradient Descent on Support Vector Machines was used. On comparing the result from the Table IV(a) and Table IV(b), we get that the SVM model performs well on both features. The SVM model using TF-IDF features performs the best, as can be observed, because of its high level of accuracy. 96.3 percent by displaying the graph in Fig. 2. For both features, several classifiers such as MultinomialNB, Decision Tree, and Random Forest are used. On the bag of a word, these classifiers exhibit (82 percent, 74 percent, and 78 percent) accuracy, using tf-idf feature the classifiers give an accuracy of MultinomialNB, decision tree, and random forest (74 percent, 76 percent, and 78 percent). Hence we figure out that using tf-idf features our model shows the best result in supervised learning.

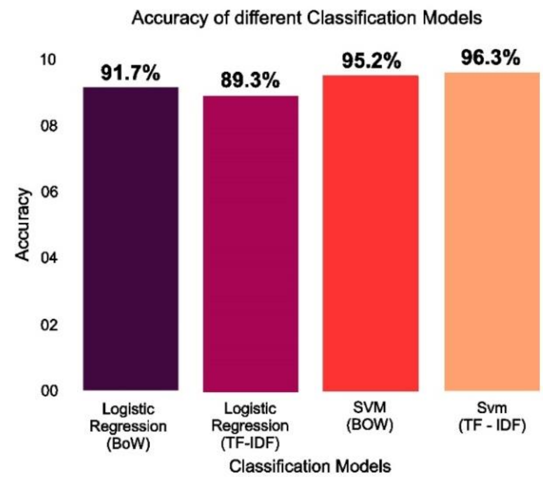

Fig. 2. Classification Model Accuracy.

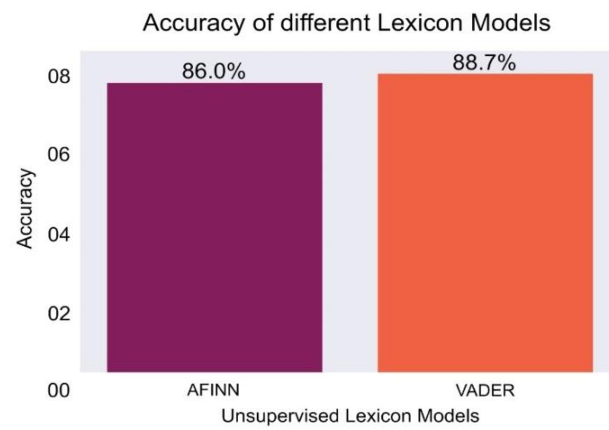

Fig. 3. Unsupervised Lexicon Model Accuracy

TABLE IV. (A). COMPARATIVE RESULT TABLE OF SUPERVISED LEARNING APPROACH USING BAG OF WORD

Class	LR-BOW				SVM-BOW			
	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F-Score	Accuracy
0	0.86	0.59	0.70	**0.917**	0.87	0.84	0.85	**0.952**
1	0.92	0.98	0.95		0.97	0.97	0.97	

(B). COMPARATIVE RESULT TABLE OF SUPERVISED LEARNING APPROACH USING TFIDF

class	LR-TFIDF				SVM-TFIDF			
	Precision	Re-call	F1-score	Occur.	precision	Re-call	F1-Score	Occur.
0	0.97	0.36	0.53	**0.893**	0.95	0.82	0.88	**0.963**
1	0.89	1.00	0.94	-	0.97	0.93	0.98	

TABLE V. COMPARATIVE ANALYSIS OF PROPOSED APPROACH PERFORMANCE OF LEXICON-BASED APPROACHES

Class	**AFINN Lexicon**				**VADER-Lexicon**			
	Precision	**Recall**	**F1-Score**	**Accuracy**	**Precision**	**Recall**	**F-Score**	**Accuracy**
negative	0.59	0.78	0.67	**0.86**	0.91	0.43	0.58	**0.887**
positive	0.95	0.88	0.91		0.89	0.99	0.93	

During the unsupervised lexicon model, we extract 30% of test data. Proposed data are used in the model. For test review, the AFINN and VADER lexicon models were utilized to generate polarity scores and accuracy. We predicted sentiment using these polarity scores in the model. Positive and negative classes are used in it. The true label is used for the test sentiment whereas the predicted label is used for the predicted sentiment, for evaluating the model's performance. We used Afinn and Vader lexicon model for sentiment and accuracy. We calculate sentiment polarity using Afinn-score over the 'Review' column. For sentiment prediction, we used sentiment polarity. Now, the predicted label shows the predicted sentiment and the true label has test sentiment. We evaluate model performance and accuracy using these labels. Model performance using precision, recall, f-measure, accuracy for both classes. Performance results are shown in the Table V. The Vader ("Valence Aware Dictionary and sEntiment Reasoner") is a lexicon and rule-based tool for sentiment analysis. SentimentIntensityAnalyzer () function takes a string and produces a dictionary of scores in positive, negative, compound, etc. categories. A compounded score is a statistic that adds up all of the lexical ratings, normalized between -1 for the most severe negative and +1 for the most extreme positive. We apply this function over 'The 'Review' column of the dataset and predict sentiment using a compound score. We evaluated model performance using the predicted sentiment and test-sentiment label. On comparing the result for lexicon models, The Vader model exceeds the lexicon Afinn models with the highest 88.7% accuracy percent, which is depicted in the graph in Fig. 3.

Now, we compare both models, the supervised and unsupervised lexicon models. In the supervised model, on comparing Table IV(a) and (b) we get that SVM outperforms using feature tf_idf with the accuracy of 96.3% which is the VADER lexicon model performs well with 88.7% accuracy. As a result, the graph in Fig. 2 depicted a significant upgrade in the value of accuracy of classifier in the supervised model, and Fig. 3 shows the accuracy of lexicons in the unsupervised model, comparing the accuracy of both models from the graphs, we can see that the supervised model outperforms the lexicon approach.

V. CONCLUSION

In this paper, we analyze sentiment using both supervised and unsupervised models. For both features BoW and TFIDF, we employed SGD-SVM and logistic regression as classifiers, with bi-gram words in the classification model whereas AFINN and Vader lexicon was used in the unsupervised lexicon model. We discovered that the Vader lexical model is 88.7 percent more accurate than other lexical models. Other models' performance on the given data is found to be comparable to VADER. In terms of Supervised Learning models, the SVM model on TF-IDF features is the best, with 96.3 percent accuracy. We may conclude that typical supervised models outperform lexicon models by equating the top models from both models. The limitation is that tone can be difficult to decipher vocally, and even more difficult to decipher in writing. Things become far more difficult when trying to analyze a huge volume of data having both subjective and objective responses.

REFERENCES

[1] D.Khurana .,A. Koli .,K. Khatter., and S. Singh .,", Natural Language Processing: State of The Art", Current Trends and Challenges",https://www.researchgate.net/publication/319164243,August 2017.

[2] H. Hassan Raza, M. Faizan, A. Hamza, A.Mushtaq .,N. Akhtar.," Scientific Text Sentiment Analysis using Machine Learning Techniques", (IJACSA) International Journal of Advanced Computer Science and applications, Vol. 10, No. 12, 2019.

[3] In European conference on machine learning (pp. 4-15). Springer, Berlin, Heidelberg, doi.org/10.1007/BFb0026666,1998.

[4] M.Ikonomakis., S.Kotsiantis, , &V.Tampakas,"Text classification using machine learning techniques" Issue 8, Volume 4, August 2005, pp. 966-974,WSEAS transactions on computers, 4(8), 966-974.

[5] Md. Taufiqul Haque Khan Tusar, Md. Touhidul Islam," A Comparative Study of Sentiment AnalysisUsing NLP and Different Machine Learning Techniques on US Airline Twitter Data", DOI:10.1109/ICECIT54077.2021.9641336,2021.

[6] Venkateswarlu Bonta, Nandhini Kumaresh, and N. Janardhan," A Comprehensive Study on Lexicon Based Approaches for Sentiment Analysis", Asian Journal of Computer Science and Technology ISSN: 2249-0701 Vol.8 No.S2, 2019, pp. 1-6.

[7] Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, Manfred Stede," Lexicon-Based Methods for Sentiment Analysis", https://doi.org/10.1162/COLI_a_00049, June 2011.

[8] S.Elbagir and J.Yang.," Twitter Sentiment Analysis Using Natural Language Toolkit and VADER Sentiment", ISBN: 978-988-14048-5-5 ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online), Proceedings of the International MultiConference of Engineers and Computer Scientists 2019 IMECS 2019, March 13-15, 2019, Hong Kong.

[9] Asghar MZ, Khan A, Ahmad S, Qasim M, Khan IA (2017) Lexicon-enhanced sentiment analysis framework using a rule-based classification scheme. PLoS ONE 12(2): e0171649. doi:10.1371.

[10] B. Das, S. Chakraborty," An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation", 2018.pdf.

[11] D.M.El-Din" Enhancement Bag-of-Words Model for Solving the Challenges of Sentiment Analysis", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 1, 2016.

[12] Jawad Khan and Young-Koo Lee, "LeSSA: A Unified Framework based on Lexicons and Semi-Supervised Learning Approaches for Textual Sentiment Classification", Received: 21 November 2019; Accepted: 13 December 2019; Published: 17 December 2019.

[13] J.Singh, G.Singh, and R.Singh," Optimization of sentiment analysis using machine learning classifiers", Singh et al. Hum. Cent. Comput. Inf. Sci. (2017) 7:32 DOI 10.1186/s13673-017-0116-3.

[14] W.Zhou, W. Hanbin, S. Hongguang, and S.Tieli," A Method of Short Text Representation Based on the Feature Probability Embedded Vector",* Received: 4 July 2019; Accepted: 26 August 2019; Published: 28 August 2019.

[15] Trip Advisor Hotel Review Datasethttps://doi.org/10.5281. zenodo.1219899.

[16] G. Forman, “An Experimental Study of Feature Selection Metrics for Text Categorization”. Journal of Machine Learning Research, 3 2003, pp. 1289-1305.

[17] Breiman, L. (2017). Classification and regression trees. Routledge.

[18] Abdulhamit Subasi, “in Practical Machine Learning for Data Analysis Using Python”, 2020.

[19] M. S. Usha, Dr. M. Indra Devi,” Analysis of Sentiments using Unsupervised Learning Techniques”, DOI: 10.1109/ICICES.2013.650820329, April 2013.

[20] C.J. Hutto, Eric Gilbert, ”VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text”,2014.

[21] Liu, B. (2010), Sentiment Analysis and Subjectivity. In N. Indurkhya& F. Damerau (Eds.), Handbook of Natural Language Processing (2nd ed.). Boca Raton, FL: Chapman & Hall.