Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2023 SCPE. Volume 24, Issues 4, pp. 847–855, DOI 10.12694/scpe.v24i4.2343

A FEATURE EXTRACTION BASED IMPROVED SENTIMENT ANALYSIS ON APACHE SPARK FOR REAL-TIME TWITTER DATA

PIYUSH KANUNGO[*]AND HARI SINGH[†]

Abstract. This paper aims to improve the accuracy of sentiment analysis on Apache Spark for a real-time general twitter data. A lot of works exist on sentiment analysis on offline or stored twitter data that uses several classification algorithms on relevant features extracted using well-known feature extraction methodologies on pre-processed text data. However, not much works exist for sentiment analysis of real-time twitter data and especially for the generic data on big data processing platforms such as Apache Spark. This paper proposes a real-time sentiment analysis for generic twitter data through Apache Spark using six classification algorithms on N-gram and Term Frequency – Inverse Document Frequency (TF-IDF) feature extraction methodologies on the pre-processed data. An exhaustive comparison is done using Logistic Regression (LR), Multinomial Naive Bayes (MNB), Random Forest Classfier(RFC), Support Vector Machine (SVM), K-Nearest Neighbour (K-NN), and Decision Tree (DT) classification algorithms. It is observed that the trigram feature extraction method performs the best on LR and SVM and the RFC results are also comparable on the considered general tweets data.

Key words: Machine learning, Apache Spark, Twitter, Sentiment analysis, N-gram, TF-IDF

1. Introduction.

Today big data applications are widely used in different fields like analysing social media platforms, improving healthcare systems, understanding customer behaviour and natural language processing. These applications play a key role for extracting the knowledge from large size datasets which can be used further for profit generation as well as for process improvement and business expansion. Social media plays a completely critical function in our life. What we are able to think, what we do, all of us express our emotions on social media platforms. Social media is a big, interactive medium for dialogue of numerous troubles associated with society in addition to vital for the growing unfolds of facts, specifically throughout instances of herbal disasters, calamities, and mass emergencies. Also, on social media humans speak about the goods which might be released day with the aid of using day. Many groups and business enterprises use those forms of facts (associated with their merchandise) to recognize what the humans reflect on consideration on their product. They can examine these facts and the usage of Social Network Analysis [1, 2]. Interactions via social media systems are now no longer centralised to a selected location, time quarter etc. Social media affords a short and effective manner to unfold facts now no longer counting whether its miles correct or inaccurate, unfolding of the both types is favourable. However social community typically favoured extra correct and legitimate facts to unfold than fake facts and rumoured facts. Interaction happens in actual time so this affords applicable unfold of facts according to applicable facts.

Understanding human sentiments and expressions from text over social media on any particular topic or event helps in better analysing and decision making. Text data mining or so called the text mining is actually bringing out the meaningful patterns and new information from the unstructured data after being processed thoroughly which leads to the structured format of the data. Features are the characteristics in the form of specific variables in any set of data that can be used to give accuracy in predictions after being selected appropriately. The increase in the web data or texts in documents have made it difficult to pre-process as the data is from many different dimensionalities. In order to reduce the data with different dimensionality the use of better feature selection and feature extraction is needed. While the feature selection or doing feature extraction depends on data being used in the application domain. Feature Extraction allows data reside in feature space without any loss of data even if the feature space size is reduced. N-gram and TF-IDF have been

[*]CSE & IT Department, Jaypee University of Information Technology, Solan, HP, India (piyushkanungo34@gmail.com)
[†]CSE & IT Department, Jaypee University of Information Technology, Solan, HP, India (hsrawat2016@gmail.com)

extensively used in the literature for feature extraction from the text.

Further, as the texts from the social media are huge and in real time so it's fast processing in real-time requires real-time big data processing frameworks. The processing capability of big data processing frameworks such as Apache Hadoop is well proven [3, 4, 5, 6, 7, 8, 9]. Apache Spark's ability to help in processing large data to achieve standardized data leverages the scalability and performance of Apache Spark to process and analyse the twitter dataset in real-time efficiently. A large number of big data processing technologies have evolved. A diversified domain of big data and technologies is presented in [10]. A detailed analysis of Apache Hadoop and Spark for big data processing is presented [11].

The goal of this paper is to develop a real-time sentiment analysis model using Apache Spark and classification algorithms on feature extracted through feature extraction methods N-Gram and TF-IDF that can accurately classify tweets as positive or negative based on the sentiment expressed in the text. The results provide insights into the sentiment of twitter users on a particular topic or event and demonstrate the effectiveness of sentiment classification on Apache Spark and feature extraction methologies for processing and analysing large volumes of social media data.

The rest of the paper is structured as follows. Section-2 describes related work. Section-3 describes the proposed sentiment analysis methodology. The results obtained are discussed in Section 4. The conclusions and future scope is presented in Section 5.

2. Related Work. Many researchers have recently applied sentiment analysis and word frequency techniques to classify people's attitudes from tweets. Understanding human expressions from textual data has been a focal point of studies and innovations for the past decade.

Twitter sentiment analysis using Naive Bayes algorithm shows an approach where various data cleaning and preparation methods are developed to make tweets more understandable in the word processing process [12]. The concept of control machine learning is used since each sample dataset consists of a pair of tweets and thoughts. The main goal is to find more effective tweet analytics. Sentiment analysis on twitter divides tweets into positive and negative classes. They are able to build a model with an accuracy of 94%. The authors applied classification algorithms directly on all the features of pre-processed data and no relevant feature extraction technique is used. In a similar work, the authors proposed a classification task in order to obtain the sentiment behind the polarity of an economic text using DT, gradient boost, naive bayes, RFC, K-NN, eXtreme Gradient Boost, SVM, and LR [13]. It was found that classifying the three groups (positive, negative and neutral), the support vector classifier performed best up to 77% accuracy on the test dataset. However, it was trained on an economic dataset which create a bias towards words which are used in economic texts and cannot be used with generic data. In a similar work based on K-Means clustering, sentiments of each cluster is analyzed across the several aspects of the Covid-19 pandemic [14]. In a similar work, the authors proposed a classifier using Machine Learning (ML) techniques that can predict the polarity of a comment [15].

In a feature extraction based work, a system is proposed where the authors applied N-gram bag of words feature extraction with different machine learning models [16]. Several models are developed on applying unigram, bigram and trigram to analyse economic texts. The models are evaluated on the metric of accuracy, recall, f1-score and precision. The data used was review datasets of Amazon, IMDB and yelp. It was observed that SVM performed the best with N-gram feature extraction method with an accuracy of 82%. A significant increase was observed in performance after applying the N-gram feature extraction technique. In another research work, the authors used two classification methods: unigrams and bigrams, and attempts were made to include bigrams in vectors to improve accuracy [17]. Once removed, the function returns as a small or dense vector. Based on the data, there is a sparse vector representation that is more efficient. The drawback of this approach is that it is just going for unigram and bigram feature extraction which many of the times are not enough to capture the sentiment as the bag of words is too small.

A sentiment analysis of hotel reviews using N-gram and Naive Bayes Methods aims to determine the application of N-gram and Naive Bayes methods in sentiment analysis [18]. Based on accuracy results, it was found that tokenization unigram method works better than other tokenization methods. This method obtained precision results of 94%, recall 100%, accuracy 97%, and error rate 3%. However, a single classification algorithm was used and as it provided best result using unigram, it can also be validated over other data. Similarly, sentiment analysis based on N-gram and K-NN Classifier [19] was applied to classify data into positive, negative and neural classes. The accuracy of proposed system is achieved up to 86%.

An N-gram feature extraction based sentiment classification model for drug user reviews using Naive Bayes, Maximum Entropy, and SVM is presented [20]. It was found that the Maximum Entropy method achieves the best result for the presence and frequency of unigram features, and the SVM method achieves the best result for TF-IDF of unigram bigram features. However, it focuses on text reviews from online health information services and hence cannot be used on generic dataset. Similarly, a sentiment classification using N-gram and TF-IDF uses a general machine learning framework over three types of document in the dataset, sentences in questions and answers on Stack Overflow; reviews of mobile applications; and comments on Jira issue trackers, is used. The method achieved highest F1 values in positive and negative sentences on all datasets in comparison to the publicly available datasets [21]. In another work, the authors use N-gram and TF-IDF feature extraction for online fake news detection on six machine learning classification models and found that the TF-IDF with SVM provides highest accuracy [22].

In another work, the authors predicted election result by enhanced sentiment analysis on twitter data using ensemble classifier and Natural Language Processing (NLP). The NLP based approach is used to enhance the sentiment classification by adding semantics in feature vectors and thereby using ensemble methods for classification [23]. Adding semantically similar words and context-sense identities to the feature vectors increased the accuracy of prediction. The comparison of experiment results show that the semantics-based feature vector with ensemble classifier outperforms the traditional bag-of-words approach with single machine learning classifier. The ensemble method performs better than the other traditional classification by 3- 5%. In a recent work, a dual-channel attention network model is used to extract the text semantic through transducive learning and graph structure to enhance the semantic classification [24]. In another recent work, the authors used the sentiment analysis of social media tweets for extracting features based on sentiment lexicons and textual content for inputting to classifier for detecting depression[25]. In another work, an orthographic pleonasm method is used for improving lexical sentiment analysis accuracy to identify emotion-related neologisms in social media texts [26] and a cross lingual sentiment classification is presented in [27]. In another work, the authors made a comparison among bag of words with Naive Bayes, continuous bag of words with SVM, and Long Short-Term Memory (LSTM) for intent classification for dialogue utterances [28]. In another work, a sentiment classification for cryptocurrency related social media tweets use the bidirectional encoder representation from transformers (BERT) model to learn the numerical representation of text and emojis based sentiment classification generates emoji sentiment lexicon – language-universal cryptocurrency emoji (LUKE) lexicon [29].

In some of the works, the Apache Spark big data processing framework is used for sentiment analysis. The authors used MLib, an Apache spark machine learning library, to classify tweets into sentiment classes [30]. They used spark's ability to handle large amount of data which helped in performing pre-processing and standardization for the huge dataset at once. They applied Naive Bayes, Logistic regression, and Decision tree with unigram and TF-IDF feature extraction method. They applied five-fold cross validation approach to ensure a precise accuracy score. They observed that the accuracy scores for naive bayes and logistic regression were close and were far superior to the decision tree. It was found that for a large dataset, 78% accuracy was achieved for logistic regression and Naive-Bayes whereas the decision tree showed an accuracy of 68%. In another Apache Spark based coronavirus pandemic prediction in real-time uses machine learning for features extracted on twitter big data streaming [31].

A sentiment classification using paragraph vector and cognitive big data semantics used Apache Spark for a distributed version of any machine learning algorithm which scales easily to larger datasets on a set of hardware devices [32]. Here, a hybrid of sentence vectors, distributed and balanced versions of well-known machine learning techniques for emotional analysis is used. They used two methods for comparison - bag of words based Document Term Matrix (DTM) and the hash trick-based DTM. The model resulted in an area under the curve (AUC) of 95.44%. The model uses DTM and did not use other feature extraction methods.

An entropy-based evaluation for sentiment analysis of stock market prices on twitter data is used [33]. Initially, the daily twitter posts are analyzed and different N-grams along with two strategies that are utilized to increase the accuracy of the classification are applied. A Spark streaming has been employed for the processing of Twitter data, while Apache Flume has been utilized for the analysis. The cons of this approach is that sentiment analysis without using appropriate machine learning is not effective and models become too simple

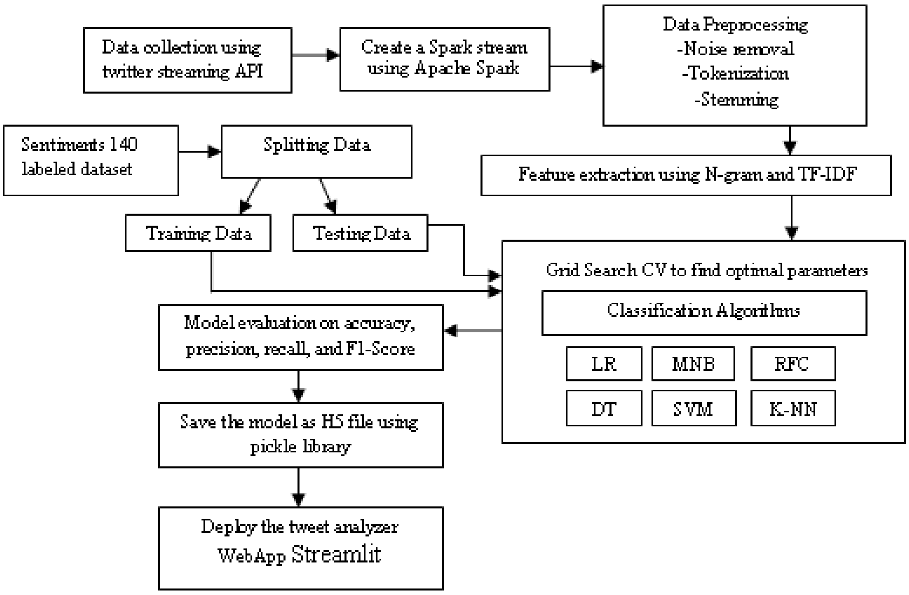
Data collection using twitter streaming API
Create a Spark stream using Apache Spark
Data Preprocessing -Noise removal -Tokenization -Stemming
Sentiments 140 labeled dataset
Splitting Data
Feature extraction using N-gram and TF-IDF
Training Data
Testing Data
Grid Search CV to find optimal parameters
Classification Algorithms
LR
MNB
RFC
DT
SVM
K-NN
Model evaluation on accuracy, precision, recall, and F1-Score
Save the model as H5 file using pickle library
Deploy the tweet analyzer WebApp Streamlit

Fig. 3.1: Proposed Methodology for Twitter sentiment analysis

to capture the correct sentiments.

This paper presents a Apache Spark based real-time sentiment analysis for general tweets data. The three main components of our work are data extraction, processing, and modelling. We utilise the Natural Language Toolkit (NLTK) dataset to create our model. The real-time text data from twitter is pre-processed for noise removal, tokenization and stemming. Then feature extraction using N-gram and TF-IDF is applied to the pre-processed data and then the extracted features are fed to the classifiers. We compared six classifiers to categorise our tweets into positive and negative attitudes based on a supervised probabilistic machine learning algorithm, and then effectiveness of the classifiers are assessed. The presented work compares six classification algorithms LR, MNB, RFC, SVM, K-NN and DT on Unigram, Bigram, Trigram and TF-IDF based features for sentiment classification.

3. Proposed Sentiment Analysis Methodology. The proposed system can identify the sentiment behind users recent tweets, and to achieve that an offline NLP model is built using Apache spark text pre-processing tools and machine learning classification algorithms, LR [34, 35], MNB [36], RFC [37], SVM, RF [38], and DT and K-NN[12] with different hyper-parameters tuning and feature extraction technique combinations to find the optimal model. The FIG. 3.1. illustrates the different stages of the system which include dataset collection, creating a spark stream, data pre-processing, data splitting, hyper-parameter tuning, feature extraction, model training, model evaluation and model deployment. Each step in building the system is explained as follows:

3.1. Data collection and creating a Spark stream. The dataset used in this paper is the sentiment140 dataset from Kaggle [39], which was used to perform a twitter sentiment classification using distant supervision [40]. The dataset has six fields which are target, ids, date, flag, user, text with 1.6 million tweet instances labelled as 0 (positive) or 4 (negative.). Upon relabelling the target values as 0 and 1 and dropping all the fields except ‘text’ and ‘target’, it is observed that the dataset is balanced and has approximately the same number of data points for both the labelled observations. We start by creating a spark session to load our dataset to stream the data so that further operations can be done. Apache spark has a lot of utility functions to perform text pre-processing and standardisation. After loading the data, exploratory data analysis is done to better understand the data so as to perform correct operations after handling missing values [41].

3.2. Data pre-processing. Data pre-processing is important in any text-based analytics system because the complexity of the data directly influences the model training when trying to extract sentiments out of textual data. According to our studies, Twitter is a platform with many links, hashtags, special characters, emojis, etc. It is considered one of the most popular text dumps of human language in digital textual form due to its content. Therefore, pre-processing removes anything from the tweet that does not add to the meaning of the tweet or just makes the text vague. So, twitter data pre-processing is performed with the following steps using NLTK library: noise removal, tokenization and stemming, which are described below:

Noise removal During this phase the useless parts of text which add no context to the tweet are removed which is achieved in the following steps:

- **Remove accented char** Accented characters generally are used to add the sense of sound to text which is not required for a text NLP model.
- **Remove emails** Email ids don't add meaning to the text so they are removed.
- **Remove hashtags** The twitter hashtags are used to index topics on the platform but are generally repetitive to and are removed.
- **Remove HTML tags** Remove unnecessary URL links using the beautiful soap library as they don't add context to the text.
- **Remove retweets** Redundant data must be removed to maintain the data unbiased and train a good model.
- **Remove multiple spaces** Multiple continuous space must be removed to avoid confusion while training the model.
- **Remove stop words** These are the insignificant parts of the human vocabulary which are basically connecting words or ending words. These are filtered words which include punctuations, articles, conjunctions and general words such as a, the, an.

Tokenization Tokenization in pre-processing text data includes dividing longer strings into tokens. These tokens can be sentences that can be broken down into short sentences, which can be broken down into words. This process makes it easier for the model to handle the complex text data.

Stemming After the tokenization stage, the next stage is stemming. During this stage the words are changed to their original form (i.e., root form to have less redundant meaning attached to similar words). For example, ”tired” and ”tiring” will be reduced to the word ”tire”.

3.3. Data splitting, model training, and evaluation. During this stage, the processed labelled dataset with 'text' and 'target' fields is split into a training and test dataset as per the stratified 10-fold CV. The training datsset is used to train the models and the test dataset is used to finally evaluate the built models. The machine learning models applied to the extracted features are as follows: LR, MNB, RFC, SVM, DT and K-NN. After loading the models, a classifier is built for each model with each type of feature extraction i.e., unigram, bigram, trigram and TF-IDF with testing set and stratified 10-fold cross-validation (CV) to find the cross- validation performance of the applied models. The standard classification report with accuracy, precision, recall, and F1-score is used to check the test performance and cross-validation performance.

3.4. Hyperparameter tuning. Now that we have the training set to start building and testing out different models, we need to decide which model specific parameters we need to pass to obtain the best results for that specific model. We decided to do that using GridSearchCV which is an algorithm which helps us to iterate through all possible parameter combinations and gives attached scores for each iteration that helps us to choose the best combination of parameters for each model. For example, we got C = 1 and kernel = 'linear' for SVM and n = 1 and weight = uniform for K-NN.

3.5. Feature extraction. When dealing with high dimensional data, feature extraction is needed to perform textual data analysis. Feature extraction basically entails converting text data into a matrix of features. N-grams and TF-IDF feature extraction techniques are applied over the text in this paper. N-gram feature extraction method is a prominent bag of words-based analysis technique used in text mining and natural language processing. An N-gram is used to calculate a contiguous series of words of length 'N' in a particular time frame according to textual data analysis. In this paper, the N-gram approach to capture the context of Twitter data uses N = 1 to N = 3 (i.e., unigram, bigram and trigram). The TF-IDF is a feature extraction method which works by identifying the most important words in a document by calculating the relative frequency of

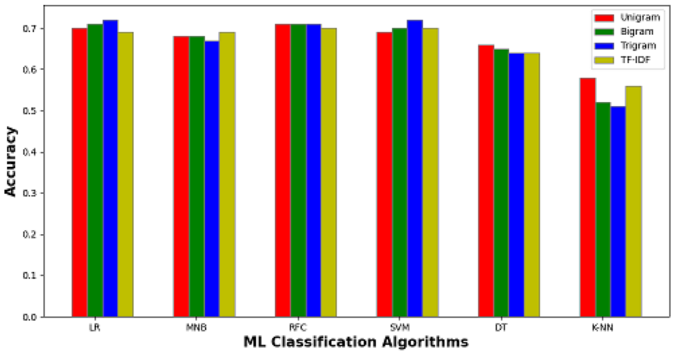

Fig. 3.2: Sentiment classification performance comparison of various feature extraction methods on ML algorithms

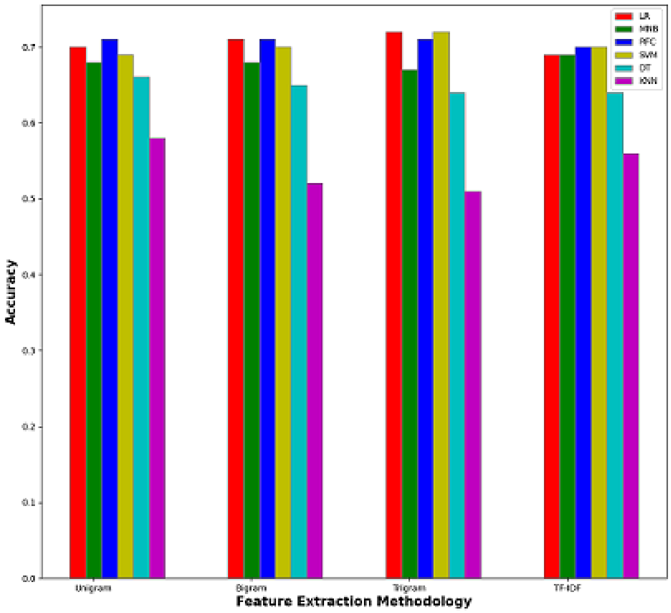

Fig. 3.3: Sentiment classification performance comparison of various ML classification algorithms on feature extraction methods

that word in the document. The main advantage of using TF-IDF is the ease of use for large document corpus but at the same time it is sometimes too simple to get hold of complex multidimensional data.

4. Results and discussions. The classification report is analysed to check the test accuracy and the cross validation accuracy to compare the performance and consistency of the models. The K-NN shows the lowest accuracy across every model for every feature extraction method i.e., 55% in case of unigram, 51% in case of

Table 4.1: Results of various ML classification algorithms on various feature extraction methodologies over the real-time twitter tweets

ML Algorithms	**Feature Extraction Methodology**	**Testing Performance**				**Cross-Validation Performance**
		Accuracy	**Precision**	**Recall**	**F1-Score**	**Accuracy**
LR	Unigram	0.7	0.71	0.7	0.71	0.64
	Biigram	0.7	0.72	0.7	0.71	0.63
	Trigram	0.72	0.74	0.71	0.72	0.64
	TF-IDF	0.69	0.68	0.69	0.7	0.64
MNB	Unigram	0.68	0.66	0.56	0.71	0.62
	Bigram	0.68	0.67	0.71	0.69	0.61
	Trigram	0.67	0.68	0.68	0.68	0.60
	TF-IDF	0.69	0.67	0.76	0.71	0.64
RFC	Unigram	0.71	0.72	0.72	0.72	0.64
	Bigram	0.71	0.74	0.66	0.7	0.64
	Trigram	0.71	0.74	0.66	0.7	0.64
	TF-IDF	0.7	0.72	0.69	0.7	0.63
SVM	Unigram	0.69	0.7	0.7	0.7 0	0.61
	Bigram	0.7	0.72	0.68	0.7	0.61
	Trigram	0.72	0.73	0.71	0.72	0.61
	TF-IDF	0.7	0.72	0.69	0.7	0.63
DT	Unigram	0.66	0.68	0.63	0.65	0.59
	Bigram	0.65	0.68	0.62	0.65	0.57
	Trigram	0.64	0.66	0.62	0.64	0.6
	TF-IDF	0.64	0.66	0.61	0.63	0.59
K-NN	Unigram	0.58	0.56	0.78	0.66	0.55
	Bigram	0.52	0.52	0.90	0.66	0.51
	Trigram	0.51	0.51	0.97	0.67	0.50
	TF-IDF	0.56	0.54	0.89	0.67	0.52

bigram, 50% in case of trigram, and 50% when TF-IDF is used for 10-fold cross validation data. Meanwhile, the LR with trigram and SVM with trigram display the highest accuracy score of 72% for the testing data and the accuracy of the RFC is also comparable. When considering different feature extraction methods trigram appears to be the most consistent approach while dealing with our dataset. It is shown in the Table 4.1 and FIG. 3.2. It displays a test accuracy of 72% for LR, 67% for MNB, 71% for RFC, 72% for SVM, 64% for DT, and 51% for KNN. It is evident from the Table 4.1 that the trigram is the best feature extraction method for the used dataset.

It is observed from the Table 4.1 and Fig. 3.3 that the LR and SVM demonstrate highest accuracy and the RFC is also comparable, which seems reasonable when dealing with binary classes namely positive and negative. And it is well known that the logistic regression performs well for binary classification. The SVM has been used historically for complex pattern recognition problems such as handwriting recognition, email classification and gene classification and therefore as expected it performs well with the dataset used as the twitter data is generic and complex to deal with. As for the feature extraction methods trigram method resulted in best observations as it is hard to capture the patterns on textual data which is generic in nature by having N = 1, 2 i.e., taking 1 or 2 bag of words at a time. It is also observed that the K-NN shows worst performance when compared to every other trained model, because when dealing with a large dataset the K-NN fails in capturing the sentiment tendencies as the K-NN is generally weak for larger datasets.

5. Conclusions and Future Work. The aim of this paper is to find the best combination of feature extraction methods and machine learning algorithms to perform real-time sentiment analysis on the Apache Spark. This paper demonstrates Apache Spark's data processing capabilities combined with feature extraction techniques when used with a huge and generic dataset and machine learning classification algorithms to get detailed observational data which can be used to compare and contrast each of the algorithms. From the experimental results, it can be concluded that according to the dataset used which is a generic tweets twitter dataset, the SVM and LR with trigram make the best classification of real-time tweets and the RFC is also comparable.

There are several other feature extraction methodologies in existence from the recent research work. We plan to extend our work for those feature extraction methodologies. Some advanced classification algorithms from the deep learning domain can be worked upon. Lastly, the real-time sentiment analysis can be applied to the new or latest prevailing text over the social media.

REFERENCES

[1] D. Verma, H. Singh, and A. K. Gupta, *A study of big data processing for sentiments analysis*, in Large-Scale Data Streaming, Processing, and Blockchain Security, IGI Global, 2020.

[2] T. Dilesh, M. Dudhane, A. Sardar, K. Deshpande, and N. Deshmukh, *Sentiment Analysis on Social Media for Emotion Classification*, 2020, doi: 10.1109/ICICCS48265.2020.9121057.

[3] H. Singh and S. Bawa,*A Survey of Traditional and MapReduceBased Spatial Query Processing Approaches*, ACM SIGMOD Record, Vol. 46, No. 2, pp. 18–29, 2017, doi: 10.1145/3137586.3137590.

[4] H. Singh and S. Bawa,*A mapreduce-based efficient H-bucket PMR quadtree spatial index*, Computer System Science and Engineering, Vol. 32, No. 5, pp. 405–415, 2017.

[5] H. Singh and S. Bawa,*Scalability and Fault Tolerance of MapReduce for Spatial data*, Glob. J. Eng. Sci. Res. Manag., Vol. 3, No. 8, pp. 97–103, 2016.

[6] H. Singh and S. Bawa,*IGSIM: An improved integrated Grid and MapReduce-Hadoop architecture for spatial data: Hilbert TGS R-Tree-based IGSIM*, Concurrency Computation : Practice and Experience, John Wiley & Sons, Vol. 31, Iss. 17, 2019, doi:https://doi.org/10.1002/cpe.5202.

[7] M. Mittal, H. Singh, K. Paliwal, and L. M. Goyal,*Efficient Random Data Accessing in MapReduce*, in International Conference on Infocom Technologies and Unmanned Systems (Trends and Future Directions), pp. 552–556, 2017.

[8] H. Singh and S. Bawa,*An Integrated Architecture for High Performance Spatial Data Analysis*, International Journal of Computer Science and Information Security, Vol. 14, No. 11, pp. 302–309, 2016.

[9] H. Singh and S. Bawa,*A MapReduce-based scalable discovery and indexing of structured big data*, Future Generation Computer Systems, Vol. 73, No. August 2017, pp. 32–43, 2017.

[10] H. Singh, R. Vasuja, and R. Sharma, *A Survey of Diversified Domain of Big Data Technologies*, Adv. Parallel Comput., Vol. 29, No. September, pp. 1–27, 2018, doi: 10.3233/978-1-61499-814-3-1.

[11] P. Sewal and H. Singh, *A Critical Analysis of Apache Hadoop and Spark for Big Data Processing*, in 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), Oct. 2021, pp. 308–313, doi: 10.1109/ISPCC53510.2021.9609518. .

[12] P. Mishra, S. A. Pati, U. Shehroj, P. Aniyeri, and T. A. Khan, *Twitter Sentiment Analysis using Naive Bayes Algorithm*, in 2022 3rd International Informatics and Software Engineering Conference (IISEC), 2022, pp. 1–5, doi: 10.1109/IISEC56263.2022.9998252.

[13] O. E. Ojo, A. Gelbukh, H. Kalvo, O. O. Adebanje, and G. Sidorov, *Sentiment Detection in Economics Texts*, in Mexican International Conference on Artificial Intelligence, pp. 271–281, 2020.

[14] V. La Gatta, V. Moscato, M. Postiglione, and G. Sperli,*COVID-19 Sentiment Analysis Based on Tweets*, IEEE Intelligent Systems, Vol . 38, No. 3, pp. 51-55, 2023, doi: 10.1109/MIS.2023.3239180.

[15] F. Jemai, M. Hayouni, and S. Baccar, *Sentiment Analysis Using Machine Learning Algorithms*, in 2021 International Wireless Communications and Mobile Computing (IWCMC), 2021, pp. 775–779, doi: 10.1109/IWCMC51323.2021.9498965.

[16] O. E. Ojo, A. Gelbukh, H. Calvo, and O. O. Adebanji, *Performance Study of N-grams in the Analysis of Sentiments*, J. Niger. Soc. Phys. Sci., Vol. 3, No. 4, pp. 477–483, 2021, doi: https://doi.org/10.46481/jnsps.2021.201.

[17] O. R. S. Mohana, S. Kalaiselvi, K. Kousalya, P. Mohamed Hanif, D. Lohappriya, and K. Khalid Ali Khani, *Twitter based sentiment analysis to predict public emotions using machine learning algorithms*, 2021, doi: 10.1109/ICIRCA51532.2021.9544817.

[18] T. Widiyaningtyas, I. A. E. Zaeni, and R. Al Farisi, *Sentiment Analysis Of Hotel Review Using N-Gram And Naive Bayes Methods*, 2019, doi: 10.1109/ICIC47613.2019.8985946.

[19] S. Bhardwaj and J. Pant, *Sentiment Analysis Approach based N-gram and KNN Classifier*,Int. J. Res. Electron. Comput. Eng., Vol. 7, No. 2, pp. 2108–2111, 2019.

[20] J. A. Kumar, S. Abirami, and T. E. Trueman, *An N-Gram Feature-Based Sentiment Classification Model for Drug User Reviews*, in Artificial Intelligence and Evolutionary Computations in Engineering Systems, 2021, pp. 277–297, doi: 10.1007/978-981-16-2674-6_22.

[21] R. Maipradit, H. Hata, and K. Matsumoto, *Sentiment Classification Using N-Gram Inverse Document Frequency and Automated Machine Learning*, IEEE Softw., Vol. 36, No. 5, pp. 65–70, 2019, doi: 10.1109/MS.2019.2919573.

[22] H. Ahmed, I. Traore, and S. Saad. *Detection of online fake news using N-Gram analysis and machine learning techniques*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 10618 LNCS, pp.127-138, 2017.

[23] R. Jose and V. S. Chooralil, *Prediction of election result by enhanced sentiment analysis on twitter data using classifier*

ensemble Approach, 2016, doi: 10.1109/SAPIENCE.2016.7684133.

[24] K. Dong, Y. Liu, F. Xu, and P. Liu, *DCAT: Combining Multi-semantic Dual-channel Attention Fusion for Text Classification*, IEEE Intelligent Systems, Vol . 38, pp. 10-19, 2023, doi: 10.1109/MIS.2023.3268228.

[25] R. Chiong, G. S. Budhi, S. Dhakal, and E. Cambriai,*Combining Sentiment Lexicons and Content-Based Features for Depression Detection*, IEEE Intelligent Systems, Vol . 36, No. 6, pp. 99-105, 2021, doi: 10.1109/MIS.2021.3093660.

[26] M. Thelwall and E. Cambriai,*This! identifying new sentiment slang through orthographic pleonasm online: Yasss slay gorg queen ilysm*, IEEE Intelligent Systems, Vol . 36, No. 4, pp. 114-120, 2021, doi: 10.1109/MIS.2021.3093660.

[27] A. Esuli, A. Moreo, F. Sebastiani, and E. Cambria,*Cross-Lingual Sentiment Quantification*, IEEE Intelligent Systems, Vol . 35, No. 3, pp. 106-114, 2020, doi: 10.1109/MIS.2020.2979203.

[28] J. Schuurmans and F. Frasincar,*Intent Classification for Dialogue Utterances*, IEEE Intelligent Systems, Vol . 35, No. 1, pp. 82-88, 2020, doi: 10.1109/MIS.2019.2954966.

[29] M. Kulakowski and F. Frasincar,*Sentiment Classification of Cryptocurrency-Related Social Media Posts*, IEEE Intelligent Systems, Vol . 38, No. 4, pp. 5-9, 2023, doi:10.1109/MIS.2023.3283170.

[30] H. Elzayady, K. M. Badran, and G. I. Salama, *Sentiment Analysis on Twitter Data Using Apache Spark Framework*, In Proceedings - 2018 13th International Conference on Computer Engineering and Systems, ICCES 2018, 2018, pp. 171–176, doi: doi: 10.1109/ICCES.2018.8639195.

[31] X. Zhang, H. Saleh, E. M. G. Younis, R. Sahal, and A. A. Al, *Predicting Coronavirus Pandemic in Real-Time Using Machine Learning and Big Data Streaming System*, Complexity, Vol. 2020, 2020, doi: 10.1155/2020/6688912.

[32] K. Ravi, V. Ravi, and B. Shivakrishnal, *Sentiment Classification Using Paragraph Vector and Cognitive Big Data Semantics on Apache Spark*, in 2018 IEEE 17th International Conference on Cognitive Informatics & Cognitive Computing (ICCI*CC), 2018, pp. 187–194, doi: 10.1109/ICCI-CC.2018.8482085.

[33] A. Kanavos, G. Vonitsanos, A. Mohasseb, and P. Mylonasl, *An Entropy-based Evaluation for Sentiment Analysis of Stock Market Prices using Twitter Data*, in 2020 15th International Workshop on Semantic and Social Media Adaptation and Personalization, 2020, pp. 1–7, doi: 10.1109/SMAP49528.2020.9248440.

[34] D. G. Kleinbaum and M. Klein, *Logistic Regression: A self learning text*, Third. Atlanta, USA: Springer, 2010.

[35] H. Singh and S. Bawa, *Predicting COVID-19 statistics using machine learning regression model: Li-MuLi-Poly*, Multimed. Syst., Vol. 28, pp. 113–120, 2022, doi: 10.1007/s00530-021-00798-2.

[36] P. P. Surya and B. Subbulakshmi, *Sentimental Analysis using Naive Bayes Classifier*, in 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), 2019, pp. 1–5, doi: 10.1109/ViTECoN.2019.8899618.

[37] M. Zhu, *Random Forest classifier for remote sensing classification*, Int. J. Remote Sens., Vol. 26, No. 1, pp. 217–222, 2005.

[38] M. Basheri, M. J. Iqbal, and A. Rahim, *Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection*, IEEE Access, Vol. 6, pp. 33789–33795, 2018.

[39] M. M. KazAnova, *Sentiment140 dataset with 1.6 million tweets*, Kaggle, 2017. https://www.kaggle.com/datasets/kazanova/sentiment140 (accessed Nov. 13, 2022).

[40] A. Go, R. Bhayani, and L. Juang, *Twitter sentiment classification using distant supervision*, 2009. [Online]. Available: https://www-cs-faculty.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf.

[41] B. Yan, Z. Yang, Y. Ren, X. Tan, and E. Liu, *Microblog Sentiment Classification Using Parallel SVM in Apache Spark* , 2017, doi: 10.1109/BigDataCongress.2017.43.

Edited by: Rajni Mohana
Special issue on: Sentiment Analysis and Affective computing in Multimedia Data on Social Network
Received: Jun 11, 2023
Accepted: Sep 26, 2023