Journal of Information Technology Management

Print ISSN: 2008-5893

Online ISSN: 2423-5059

ML Based Social Media Data Emotion Analyzer and Sentiment Classifier with Enriched Preprocessor

Jayamalini Kothandan

Research Scholar, Computer Science Engineering, Bharath University, Chennai, India. E-mail: malini1301@gmail.com

Ponnavaikko Murugesan

Provost, Bharath University, Chennai, India. E-mail: ponnav@gmail.com

Abstract

Sentiment Analysis or opinion mining is NLP's method to computationally identify and categorize user opinions expressed in textual data. Mainly it is used to determine the user's opinions, emotions, appraisals, or judgments towards a specific event, topic, product, etc. is positive, negative, or neutral. In this approach, a huge amount of digital data generated online from blogs and social media websites is gathered and analyzed to discover the insights and help make business decisions. Social media is web-based applications that are designed and developed to allow people to share digital content in real-time quickly and efficiently. Many people define social media as apps on their Smartphone or tablet, but the truth is, this communication tool started with computers. It became an essential and inseparable part of human life. Most business uses social media to market products, promote brands, and connect to current customers and foster new business. Online social media data is pervasive. It allows people to post their opinions and sentiments about products, events, and other people in the form of short text messages. For example, Twitter is an online social networking service where users post and interact with short messages, called "tweets." Hence, currently, social media has become a prospective source for businesses to discover people's sentiments and opinions about a particular event or product. This paper focuses on the development of a Multinomial Naïve Bayes Based social media data emotion analyzer and sentiment classifier. This paper also explains various enriched methods used in pre-processing techniques. This paper also focuses on various Machine Learning Techniques and steps to use the text classifier and different types of language models.

Keywords: Machine learning, Multinomial naive bayes, Emotion analysis, Language models, Opinion Mining (OM), Sentiment Analysis (SA), Twitter.

DOI: 10.22059/jitm.2021.80614

Introduction

Social media are computer-mediated (Jayamalini, K., & Ponnavaikko, M., 2019) technologies that facilitate the creation and sharing of information, ideas, career interests, and other forms of expression via virtual communities and networks. The variety of stand-alone and built-in social media services currently available introduces challenges of definition; however, there are some common features:

User-generated content, such as text messages, multimedia contents such as digital photos or videos, and data generated through all online interactions, is the lifeblood of social media. Users create service-specific profiles for the website or mobile app designed and maintained by the social media organization. Social media facilitate the development of online social networks by connecting user's with other individuals or groups.

Sentiment Analysis or opinion mining (Rathi, et al, 2018) is NLP's method to computationally identify and categorize user opinions expressed in textual data. Mainly it is used to determine the user's opinions, emotions, appraisals, or judgments towards a specific event, topic, product, etc. is positive, negative, or neutral. In this approach, a huge amount of digital data generated online from blogs and social media websites is gathered and analyzed to discover the insights and help make business decisions.

Online social media data is pervasive. It allows people to post their opinions and sentiments about products, events, and other people in the form of short text messages. For example, Twitter is an online social networking service where users post and interact with short messages, called "tweets." Hence, currently, social media has become a prospective source for businesses to discover people's sentiments and opinions about a particular event or product.

Sentiment Analysis based on Twitter can be really useful for a variety of tasks such as predicting stock markets, opinions of a product, political outcomes, and much more. This paper focuses on ML-based system development to classify the given tweet into either positive or negative. This paper focuses on the development of a Machine learning Based social media data emotion analyzer and sentiment classifier. This paper also explains various enriched techniques of pre-processing of Text data.

Importance of Social Media Data

Social Media is important for business. Social media helps business to

a) create successful social campaigns using marketing analytics

b) recognize influencers for their brand, product, service & industry

c) compare key performance metrics and to find strengths, weaknesses of competitors using competitive intelligence

d) discover the real-time trending topics ie, what people are talking about the industry, product, brand, and customer opinions

e) Keep track of the virality of content spreads across the social media and the World Wide Web.

About Twitter

Twitter is the most popular microblogging site (Gupta, A., et al, 2019)one driven by short, textual messages or "microblogs." Twitter is the third most popular social network in the U.S. Twitter is frequently used to report, react to, and engage with topics of national and international importance. Twitter users can:

- Find and add friends
- Find and follow companies, entertainers, and more
- Create a short bio—about one sentence in length
- Share links to anything on the Web
- Use privacy settings to control information flow
- Track “trending topics
- Search for Twitter users’ sentiments and opinions

Tweet: A short, 140-character message Twitter users broadcast to their contacts. Twit/Tweeple/Tweeps: Nicknames for people who use Twitter.

Functionalities of Twitter:

- Retweet
- @Message: Public Tweets
- DM/Direct Message: private message to another Twitter individual.
- Hashtags (#s): # in front of a word, hashtags are a way to link your tweet to an index of tweets on related topics. Ex: #NYC, #reading, #worldcup, #GOP, etc.
- Unfollow: to remove a Twitter contact
- Favorite: If you like a tweet, then you can “favorite” it,
- Lists/Listed: This is a way to organize the accounts you’re following into categories.

- Trends: This is a list of the top 10 phrases used on Twitter at any given moment.
- Microblogging: The act of broadcasting short, in-the-moment textual messages sent via platforms like Twitter

Volume of Tweets

Every second, on average (Gupta, A., et al, 2019), around 6,000 tweets are tweeted on Twitter (visualize them here), which corresponds to over 350,000 tweets sent per minute, 500 million tweets per day, and around 200 billion tweets per year.

Format of Tweet

Twitter has developed its own language conventions (Rathi, M., et al, 2018). The following are examples of Twitter conventions:

a) “RT” is an acronym for retweet, which indicates that the user is repeating or reposting.

b) "#" stands for the hashtag is used to filter tweets according to topics or categories.

c) “@user1” represents that a message is a reply to a user whose user name is “user1".

d) Emoticons and colloquial expressions or slang languages are frequently used in tweets

e) External Weblinks (e.g., http://amze.ly/8K4n0t) are also frequently found in tweets to refer to some external sources.

f) Length: Tweets are limited to 140 characters.

The architecture of the Proposed System

ML-based Emotion Analyzer is used to analyze the twitter data using enriched pre-processing techniques and a multinominal Naïve Bayes Classifier. This system has been used by businesses to enhance customer experience. The framework of the proposed system is shown in the figure below. It comprises of:

- Tweets Extractor
- Enriched Pre-processor
- Feature Extractor
- Emotion Classifier
- Accuracy Finder

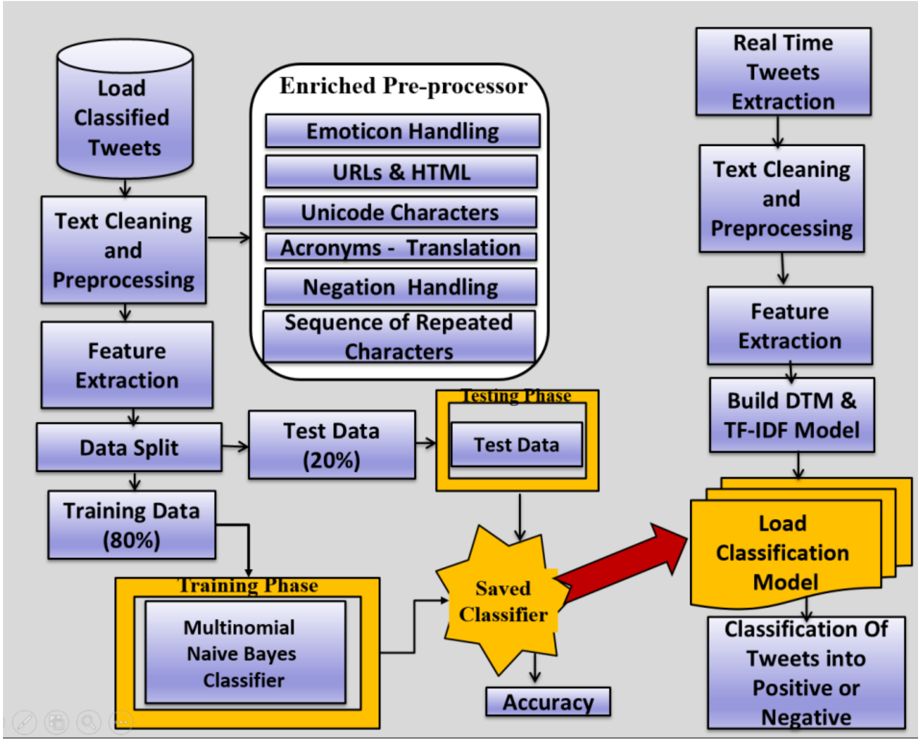

Figure1. The architecture of proposed ML Based Emotion Classifier

- Tweets Extractor: It is used to extract Tweets from Twitter after authenticating Twitter API.
- Enriched Text Cleaner and Pre-processor: It is used to convert the raw text into clean text by removing numeric values, non-English characters, URLs, white spaces, and stop words. It also handles case sensitive issues of text and stemming process.
- Feature Extractor: It is used to transform the tweets into a set of features which represent the original data without any loss of information using a dimension reduction technique.
- Emotion Classifier: It is used to find each tweet's polarity and classify them into positive or negative.
- Accuracy Finder: It is used to find the accuracy of the system.

Methods of Implementation

This system implementation is divided into two main categories:

A. Enriched Text Cleaner and Pre-processor (Billal, B., et al, 2016)

B. Emotion Classifier using Multinomial Naive Bayes (Singh, G., et al, 2019)

A. Enriched Text Cleaner and Pre-processor

a. Dataset

Tweets are slang words that are used to express users' emotions about current affairs on Twitter. The sample dataset contains around 1600000 classified tweets with four columns ItemID, Sentiment, SentimentSource, and SentimentText. The Sentiment column corresponds to the label class holding a value, 0 for the negative tweet, and 1 for the positive tweet. Tweets contain Hashtags, @username, MT (Modified Tweet), RT (Retweet), Emoticons, acronyms, and spelling mistakes.

	ItemID	Sentiment	SentimentSource	SentimentText
0	1	0	Sentiment140	is so sad for my APL friend............
1	2	0	Sentiment140	I missed the New Moon trailer...
2	3	1	Sentiment140	omg its already 7:30 :O
3	4	0	Sentiment140	.. Omgaga. Im sooo im gunna CRy. I've been at this dentist since 11.. I was suposed 2 just get a crown put on (30mins)...
4	5	0	Sentiment140	i think mi bf is cheating on me!!! T_T
5	6	0	Sentiment140	or i just worry too much?
6	7	1	Sentiment140	Juuuuuuuuuuuuuuuuussssst Chillin!!
7	8	0	Sentiment140	Sunny Again Work Tomorrow :-| TV Tonight
8	9	1	Sentiment140	handed in my uniform today . i miss you already
9	10	1	Sentiment140	hmmmm.... i wonder how she my number @-)

Figure 2. Sample Data from the Dataset

The dataset is well-balanced between negative and positive sentiment, which is depicted in the figure below:

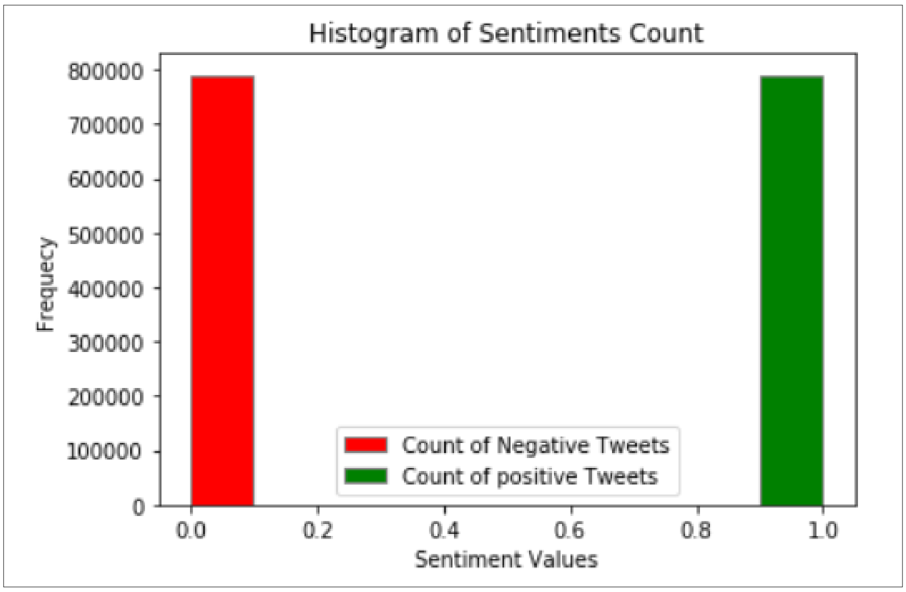

Figure3. Sample graph – count of labeled tweets

b. Other Resources

The following resources are used to facilitate the preprocessing module of our system:

- Emoticon dictionary – Contains around 132 most used emoticons in western with their sentiment value.
- Acronym dictionary - Contains 5465 acronyms with their translation.
- Stop word dictionary – Contains words that are filtered out before processing in NLP data because they do not add any value to the sentence.
- Positive and Negative word dictionaries – Contains a list of positive words (2005) and Negative words (4782).
- Negative contractions and auxiliaries dictionary – used to detect negation in a given tweet

c. Enriched Text Cleaner and Pre-processor

The data preprocessing can often have a significant impact on the performance of a supervised ML algorithm. The steps that are carried out by the enriched preprocessor of this system are as follows:

- Using the emoticon dictionary Substitute all the emoticons with their sentiment polarity value ||pos||/||neg||.
- Replace URLs with a tag ||url|| using Regular Expressions
- Removal of Unicode characters
- Decode HTML entities
- Reduce all letters to lowercase
- Replace usernames/targets @ with ||target||
- Replace acronyms with their translation
- Replace negations like not, no, never by tag ||not||
- Replace the sequence of repeated characters with two characters (e.g., "helloooo" = "helloo") to keep the emphasized usage of the word.

i. Emoticons Handling

To replace all emoticons with their corresponding polarity tags ||pos||/||neg||, the emoticon dictionary is used with a regex. The dataset consists of around 19400 positive emoticons and around 11000 negative emotions.

SentimentText
is so sad for my APL friend.............
I missed the New Moon trailer...
omg its already 7:30 ||pos||
I've been at this dentist since 11.. I was suposed 2 just get a crown put on (30mins)...
i think mi bf is cheating on me!!! ||neg||
or i just worry too much?
Juuuuuuuuuuuuuuuussssst Chillin!!
Sunny Again Work Tomorrow ||neg|| TV Tonight
handed in my uniform today . i miss you already
hmmmm.... i wonder how she my number ||pos||

Figure 4. Sentiment Text after Emoticon Handling

ii. Removal of URLs, HTML Links, and Unicode Characters

URLs, HTML links, and Unicode characters present in the sentiment text will not add any value to find the polarity. Unicode characters can cause problems during the tokenization process. Only ASCII characters are allowed in text processing.

iii. Handle Cases(to lower case)

Text messages frequently have a variety of capitalization reflecting the starting of sentences and the proper nouns. The common approach is to convert the entire text to lower case for simplicity. While changing to lowercase, it is important to remember that words like "US" to "us" can change meanings when changed to the lower case

iv. Replace all usernames/targets @ with the tag ||target||

No need to take into account usernames in order to determine the sentiment of a tweet. It should be replaced by the tag ||target||.

v. Acronyms Handling

Replace all acronyms with their translation using the acronym dictionary.

```
1) lol => laughing out loud : 59000
2) u => you : 54557
3) im => instant message : 51099
4) 2 => too : 42645
5) gonna => going to : 23716
6) 4 => for : 18610
7) dont => don't : 18363
8) wanna => want to : 16357
9) ok => okay : 16104
10) ur => your : 12960
11) omg => oh my god : 12178
12) n => and : 10415
13) ya => yeah : 9948
14) gotta => got to : 9243
15) r => are : 8132
16) tho => though : 7696
17) tv => television : 6246
18) o => oh : 6002
19) kinda => kind of : 5953
20) pic => picture : 5945
```

Figure 5. Top20 Acronyms, its Translation & count in Dataset

vi. Stop words and white spaces Handling

A word is given in the text, which is used to connect parts of a sentence rather than showing subjects, objects, or intent. A word like "the" or "and" cab be removed by comparing the text to a list of stop words.

vii. Negations Handling

We replace all negations such as not, no, don't, and so on, using the negation dictionary to take more or fewer sentences like "I don't like it." In this case, like should not be considered with positive polarity because of the "don't" present before. To do so, we will replace "don't" by ||not||, and the word "like" will not be counted as positive polarity.

When a negation word occurs, the words followed by the negation word contained in the positive and negative word dictionaries will be reversed, i.e., positive words hold negative values, and negative words hold positive values.

viii. Handling of a sequence of repeated characters

Words are emphasized using a sequence of repeated characters. The number of repeated characters to be removed to reduce the feature space.

B. Machine Learning Algorithms

Once we have completed the preprocessing part's different steps, we can now focus on the machine learning part. There are three major methods used to classify a sentence into positive or negative: SVM, Naive Bayes, and N-Gram. We focus only on Naive Bayes and N-Gram, the most commonly used methods.

i. Naive Bayes Classifier

A classifier is a machine learning model that distinguishes different objects based on certain features.

A Naive Bayes classifier is a probabilistic machine learning model that's used for classification. It works based on the Bayes theorem.

$$P(y|X) = \frac{P(X|y).P(y)}{P(X)} \tag{1}$$

The probability of **'y'** happening, given that **X's** occurrence had been calculated Using Bayes theorem. At this point, **y** is called the hypothesis**, and X** is called evidence. The hypothesis made at this point is that features are independent of each other. It means the occurrence of one specific feature does not affect the other features. For example, if there are 'n' number of features$(X_1, X_2, X_3, \ldots .X_n)$. Then X is rewritten as $X = (X_1, X_2, X_3, \ldots .X_n)$

Types of Naïve Bayes:

There are 3 types of Naïve Bayes (Singh, G., et al, 2016):

- **Multi-variate Bernoulli Modelor Binomial model**, useful if the feature vectors are binary (e.g., 0s and 1s). An application can be text classification with a bag of words model where the 0s are used to represent "words do not present in the document" and 1s are used to represent "words present in the document."
- **Multinomial Naïve Bayes**: This model is used for discrete counts. In-text classification, the Bernoulli model is extended to count the number of times the word 'wi' appears over the number of words rather than saying 0 or 1 if the word present or not.
- **Gaussian Model:** In this Model, Instead of discrete counts, it has continuous features.

The most used model for text classification is the Multinomial Naive Bayes Model.

This estimation uses the simplest smoothing method to solve the zero-probability problem that arises when the model encounters a word seen in the test set but not in the training set, Laplace, or add-one since we use one as constant.

Baseline

The Multinomial Naive Bayes classifier with Laplace smoothing represents the classic way of doing text classification. To extract features from the tweets dataset, the bag of words model is used to represent it. The bag of words model is a simplified representation of a document where it is represented as a bag of its words without taking any consideration of the grammar or word order. In-text classification, the frequency of each word is used as a feature for training a classifier.

ii. Splitting of Dataset

First, the data set should be divided into training and test set. The following steps are carried out to split the dataset:

- Shuffling of data set to avoid keeping of any order
- Separate positive and negative tweets
- divide 3/4 of the dataset into training data and 1/4 of the dataset into testing data
- Shuffle the training and test data to break the order of tweets based on their sentiment

Table 1. Training and Testing data count

Item	Count of records
Training set	1183958
Test set	394654

Validation Set

It is used to validate the model against unseen data. It is also used to tune the possible parameters of the learning algorithm to avoid underfitting and overfitting problems used to occur while training the model. The training dataset is split into two parts, 60%/ and 20%, with a ratio of 2:8 where each part contains an equal distribution of example types. The classifier will be trained with the largest amount of dataset and predict with the smaller dataset to validate the model.

K-fold cross-validation is used for validation. In this, the data set is split into k parts (k=10), hold out one, combine the others, and train on them, then validate against the held-out portion. The same process is repeated k times (each fold), holding out a different portion of data each time. Finally, average out each fold's score to find an accurate estimation of the model's performance.

Result Analysis

The accuracy of the classifier is evaluated using two methods:

- F1 score
- Confusion matrix
- F1 score

- F1 score: The F1 Score is used to measure the accuracy of a classifier, and itis calculated as a weighted average of the precision and recall. It is calculated using the given formula:

$$F1 = \frac{2 * (precision * recall)}{(precision + recall)} \quad (2)$$

F1 score lies between 0 – 1, and it reaches its best value at 1 and the worst value at 0.

Precision is the number of true positives divided by the total number of elements labeled as belonging to the positive class, and it is given by:

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

The recall is the number of true positives divided by the total number of elements that belong to the positive class, and it is given by:

If precision = 1.0, we can conclude that every result retrieved was relevant, but there is no way to find whether all relevant elements were retrieved. If recall=1.0, we can conclude that all relevant documents were retrieved, but there is no way to find twitter how many irrelevant documents were retrieved.

There is a trade-off between precision and recall where an increase in one will decrease the other. So it is advisable to use measures like F1-Score that combines precision and recall. The F1 score for each fold, then i\the values are averaged out together to find the mean accuracy on the entire training set is given in the table below:

Table 2. F1 – Score Value

Item	Values
Total Number of Tweets Classified	1183958
Mean Accuracy of F1- Score	0.776

The trained model gives an accuracy of 0.77.

- Confusion matrix

A confusion matrix is a table-like structure that is used to describe the performance of a "classifier" on a set of test data for which the true values are known. It also visualizes the performance of an algorithm. Table 3 shows the confusion matrix values predicted by the trained classifier.

Table 3. Confusion matrix

465021 (True Positive)	126305 (False Positive)
136321 (False Negative)	456311 (True Negative)

Visualization of the Confusion matrix without normalization is shown in the figure below.

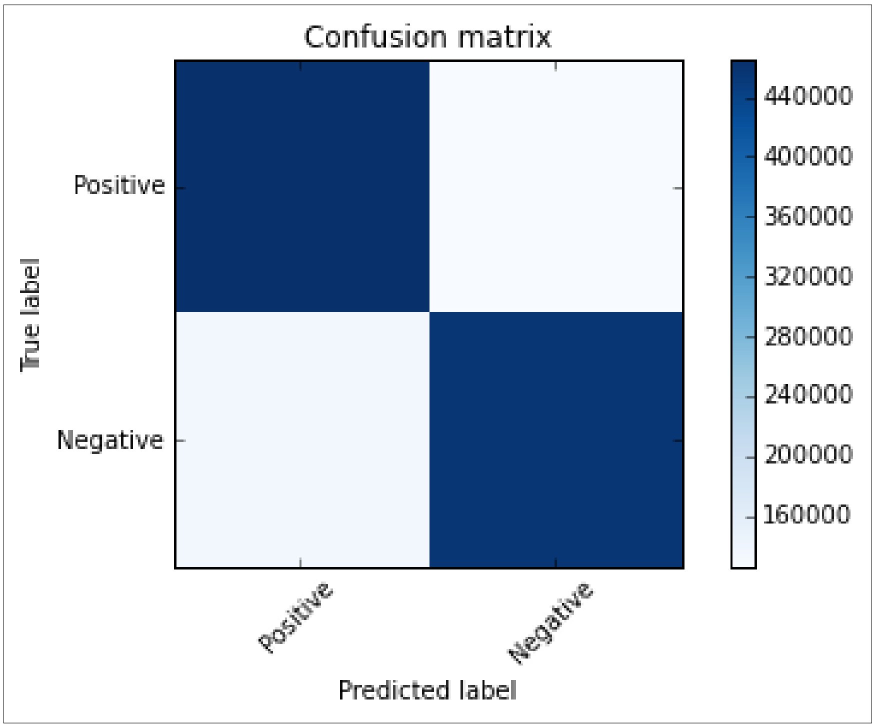

Figure 6. Confusion matrix without normalization

An N-gram language model can be applied to text classification like the Naive Bayes model to improve accuracy. Using the bigram model with the Text classifier increases the accuracy by 0.01.

Table 4. F1 – Score with Bigram Feature

Item	Values
Total Number of Tweets Classified	1183958
Mean Accuracy of F1- Score	0.784

Table 5. Confusion matrix with Bigram Feature

480120 (True Positive)	111206 (False Positive)
138700 (False Negative)	453932 (True Negative)

But using both unigram and bigram features increases the accuracy of text classifiers slightly more.

Table 6. F1 – Score with Unigram and Bigram Feature

Item	Values
Total Number of Tweets Classified	1183958
Mean Accuracy of F1- Score	0.7953

Table 7. Confusion matrix with Unigram and Bigram Feature

486521 (True Positive)	104805 (False Positive)
132142 (False Negative)	460490 (True Negative)

Conclusion

This paper focused in detail on finding what kind of emotions and sentiments expressed in tweets using enhanced preprocessor techniques and machine learning approaches. It also elaborates on the need for the large volume of free social media data available online and finding different user opinions like positive, negative, or neutral. This method of finding user opinion helps the business to create successful social operations, identify influencers for their product, service & industry, compare strengths & weaknesses of competitors, discover the real-time trending topics ie, what people are talking about the business and customer opinions & sentiment towards their business.

References

Billal, B., Fonseca, A., & Sadat, F. (2016, December). Efficient natural language pre-processing for analyzing large data sets. In *2016 IEEE International Conference on Big Data (Big Data)* (pp. 3864-3871). IEEE.

Gupta, A., Singh, A., Pandita, I., & Parashar, H. (2019, March). Sentiment Analysis of Twitter Posts using Machine Learning Algorithms. In *2019 6th International Conference on Computing for Sustainable Global Development (INDIACom)* (pp. 980-983). IEEE.

Jayamalini, K., & Ponnavaikko, M. (2019). Enhanced social media metrics analyzer using twitter corpus as an example. *Int. J. Innov. Technol. Explor. Eng.(IJITEE)*, *8*(7), 822-828.

Rathi, M., Malik, A., Varshney, D., Sharma, R., & Mendiratta, S. (2018, August). Sentiment analysis of tweets using machine learning approach. In *2018 Eleventh international conference on contemporary computing (IC3)* (pp. 1-3). IEEE.

Singh, G., Kumar, B., Gaur, L., & Tyagi, A. (2019, April). Comparison between multinomial and Bernoulli naïve Bayes for text classification. In *2019 International Conference on Automation, Computational and Technology Management (ICACTM)* (pp. 593-596). IEEE.

Bibliographic information of this paper for citing:

Jayamalini, K., & Ponnavaikko, M. (2021). ML Based Social Media Data Emotion Analyzer and Sentiment Classifier with Enriched Preprocessor. *Journal of Information Technology Management*, Special Issue, 6-20.