Multimedia Tools and Applications
https://doi.org/10.1007/s11042-023-17333-2

Comparative study of machine learning algorithms for Kannada twitter sentimental analysis

Pushpa B. Patil[1] · **Dakshayani Ijeri**[1] · **Shashikiran A. Kulkarni**[1] · **Sayed Salman Burkaposh**[1] · **Rani Bhuyyar**[1] · **Vijayalaxmi Gugawad**[1]

Received: 29 March 2022 / Revised: 29 July 2022 / Accepted: 29 September 2023

Abstract

Analyzing the client's reviews from various online platform helps to improvise the business to higher levels. These User's opinions can be analyzed using Sentiment Analysis. Sentimental analysis on Indian languages is a tedious work as there is a wide diversity in different languages of the India. Kannada is one of the prominent languages in India as 43 million of Indian population use Kannada as their native language for communication and it holds 27th rank among top 30 languages across the world, as there is very less work carried out on Indian languages, especially in Kannada language, more work is required to process the Kannada language across different domains. The sentimental analysis on the Kannada language has the accuracy about 72% from the previous work. So, in this work, we have made comparative study of various machine learning algorithms for Kannada Twitter sentimental analysis. It is experimented on live Twitter data and found that Multinomial Naive Bayes Classifier has performed better with accuracy of 75%.

Keywords Sentimental analysis · Logistic Regression Classifier · Stochastic Gradient Descent Classifier · K Neighbors Classifier · Multinomial Naive Bayes Classifier · Gaussian Naive Bayes Classifier

✉ Dakshayani Ijeri
cse.ijeri@bldeacet.ac.in

Pushpa B. Patil
cs.pushpa@bldeacet.ac.in

Shashikiran A. Kulkarni
shashikirankulkarni@gmail.com

Sayed Salman Burkaposh
salman.burkaposh@gmail.com

Rani Bhuyyar
ranibhuyyar1999@gmail.com

Vijayalaxmi Gugawad
vijayalaxmigugawad@yahoo.com

[1] Department of Computer Science and Engineering, BLDEA's V. P. Dr. P. G. Halakatti College of Engineering & Technology, Vijayapur, Karnataka, India

Published online: 24 October 2023

1 Introduction

In the last few years, there has been an exponential growth in usage of information technology. Digital technology has changed the life of human beings by providing faster way of communication with the help of internet community for instance Twitter, Facebook, WhatsApp etc. Currently almost there are more than billion active internet users. As information technology has advanced, all business has moved online such as E-commerce, Movie ticket booking and Education. So, it has become crucial thing to analyze the public's opinions in the social media to get business to higher levels. These User's opinions can be analyzed using Sentiment Analysis.

Natural Language Processing (NLP) [2] involves different steps to break the sentences and analyze the structure of a language. Sentiment analysis [1] can be achieved by using various natural language steps like tokenization, stemming, parts of speech tagging and many more.

Sentiment Analysis [1] is a Natural Language Processing [2] technique which is used to identify and classify the meaning of a writer's views such as positive or negative and can be applied on several documents. On an overview sentimental analysis helps to identify the judgement of a writer/user in terms of negative and positive reviews.

Due to the wide range of dependency on online platforms for every other requirement of people, the sentimental analysis has become the most prominent application of NLP through which the public will be able to understand the reviews given by other people. Most importantly it has become necessary to develop the system for Indian languages like Kannada. The sentimental analysis can be categorized as document-based analysis in which the whole document of text is processed and summarized to extract the meaning of document as positive or negative, sentence based in which analysis is made with every sentence and it can be phrase based in which polarity of words will be considered and identified as positive or negative. Sentiments are labeled with various human feelings with different emotions such as happiness and satisfaction are considered as positive, disappointment and feeling of low are considered as negative.

1.1 Related work

In 2021 Mandalam and Sharma [1] proposed a model for sentiment analysis on code mixed data. This approach is experimented on Tamil and Malayalam code mixed data. The proposed approach uses three methods namely sub-word level model, a word embedding based model and Machine learning based architecture. Long Short-Term Memory (LSTM) network is used by both sub-word level model and word embedding based model. Machine learning model uses term frequency-inverse document frequency (TF-IDF). The model attained final weighted F1 scores of 0.65 for Tamil and 0.68 for Malayalam.

In 2021 Madan and Ghose [2] proposed a model for sentiment analysis for twitter data in the Hindi Language. The proposed system uses two approaches of sentiment analysis. At first it uses the Lexicon Based Approach (LBA) which is based on SentiWordNet for resource. This dictionary consists of positive and negative sentiment scores attached to it through which polarity is decided for each word and based on this the sentiment is finalized. The second approach is hybrid approach which is based on both unigram and Tf-Idf model and then experimented with supervised machine learning algorithm. The first approach attained accuracy of 60.3%1 for positive text and 62.78% for negative text whereas second approach attained accuracy of 92.97% with Decision Tree algorithm.

In 2021 Kakuthota et al. [3] had developed model for sentiment analysis of tweets in Kannada, Hindi, Tamil, Telugu and Malayalam languages. The proposed system uses TextBlob package from python in which the predefined categorized words are stored based on these words the polarity of input text is calculated and through which the sentiment is classified. The system had attained accuracy of 98%.

In 2021 Kannadaguli [4] had discussed the sentiment analysis using code diverse Kannada-English dataset. The author had developed text-based Kannada-English code diverse dataset. The number of words restricted per sentence are 15 due to which the sentence is counted for first 15 words and rest words are deleted. This datafile is experimented and analyzed for sentiments with different Machine Learning algorithms methods. The algorithms used are Decision Tree, K-Nearest Neighbors, Logistic Regression, Naïve Bayes, Random Forest and Support Vector Machine. Bi-LSTM depicts better ratings with average precision of 0.54, average recall of 0.51 and average F1-score of 0.54`

In 2021 Ranjitha and Bhanu [5] proposed a model for Kannada sentiment analysis using Decision Tree Algorithm. The proposed model uses Kannada word dictionary of 500 positive and negative words each. The system uses the Decision Tree algorithm to predict the sentiment of an input sentence based on the polarity calculation and attained the accuracy of 85%.

In 2017 Hegde and Padma [6] had proposed a system for sentiment analysis of Kannada mobile product reviews using Random Forest Ensemble algorithm. The proposed approach uses the data of mobile features with 4 labels such as bad, Okok, good and best for sentiment classification and sentiment is classified using Random Forest Ensemble algorithm. The system attained the accuracy of 72%.

In 2016 Phani et al. [7] discussed the sentiment analysis approach for three Indian languages namely Hindi, Bengali and Tamil. The proposed approach used the six algorithms for sentiment analysis namely Multinominal Naïve Bayes, Logistic Regression, Decision Tree, Random Forest, Support Vector Machine (SVC) and Linear SVC. The system had used 277 positive, 354 negative and 368 neutral training tweets in Bengali language. 168 positive, 559 negative and 495 neutral tweets in Hindi language. 387 positive, 316 negative and 400 neutral training tweets in Tamil language. The development data used for Bengali language is 53 tweets, Hindi language is 56 and no data is used for Tamil language. The system attained accuracy of 67% in Bengali language, 81.57% in Hindi language and 62.16% in Tamil language.

In 2016 Rohini et al. [8] proposed a model for sentiment analysis in Kannada language using Decision Tree algorithm. The proposed approach is carried out on 100 movie reviews. The algorithm is implemented on English translated text.

In 2015 Kumar et al. [9] proposed an approach for analysis of user's sentiments from Kannada web documents. This model uses Turney's algorithm for translating Kannada reviews to English. This datafile is then experimented with sentence level approaches for analysis. Sentiment is analyzed with different machine learning methods such as J48, Random Tree, ADT Tree, Breadth First, Naïve Bayes and support Vector Machine. The algorithms are experimented on translated English reviews. The dataset consists of 182 positive and 105 negative reviews. The model achieved the average precision of 7.22% with machine learning approach.

In 2015 Hegde and Padma [10] proposed a model for sentiment analysis in Kannada language using Naïve Bayes Algorithm. The research work does not cover any information regarding size of dataset. The system attained accuracy of 65%.

In 2021 Bera et al. [11] developed a model for tweets sentiment analysis of multilingual languages. The proposed work is experimented with three algorithms such as simple neural network, convolutional neural network and long short-term memory neural network. The approach is experimented with 4,000 sample of reviews in English, Hindi and Bengali languages together and attained the accuracy of 84.1% with a model built with both long-short term neural network and convolutional neural network.

In 2020 Sharma and Ghose [12] proposed a model of twitter data sentiment analysis for general elections in India. This approach is experimented on English language. The tweeter data is considered for two candidates from January 2019 to March 2019 which consists 1967 positive words and 783 negative words. The model uses the AYLIEN package for named entity extraction and R, Rapid-Miner AYLIEN package for sentiment analysis.

In 2017 Impana and Kallimani [13] proposed an approach for sentiment analysis in Indian regional languages based on cross-Lingual method. The model uses Bilingually constrained recursive auto encoder (BRAE) for sentiment analysis of two languages. English and Kannada are the two languages considered for this work. English is resource rich language as compared to Kannada. The model had focused on construction of supervised classifier for cross lingual languages. The sentiment analysis is dominated by English language as it is the rich resource language over the Kannada language.

In 2017 Naidu et al. [14] proposed a sentiment analysis for Telugu language. The proposed model is experimented in two phases, at first it focuses to classify the sentences as subjective or objective and later it classifies the subjective sentences as positive or negative. 1400 Telugu sentences from various news papers are used for dataset. The model achieved 81% accuracy.

The system [1] was developed with an intention to improve the F1-scores as compared to existing work and it was able to achieve the improvement by a marginal value. The proposed system [2] had attained the accuracy of 92.97%, it would have been more helpful if the work mentioned regarding the data set size considered for experimentation and accuracy calculation. The proposed work [3] has used the Twitter Search API for collecting data, but this API would produce very limited number of tweets, which means the system is experimented on a limited data size. The author had mentioned the comparison of various work along with different classifiers with their respective accuracies, it would be more explanatory if the proposed work discussed regarding the classifier used and proof of accuracy calculation. The proposed model [4] contains the data set which is a combination of both Kannada and English language due to which it is hard to judge the language used for sentiment analysis, this approach has the restriction of 15 words per sentence, only first 15 words are considered and rest are deleted from a sentence which may lead to incorrect sentiment analysis and it would have been more supportive if work discusses the accuracy of the system. The system analyzes [5] sentiments of words and the accuracy of system is calculated with 1000 words. The proposed work [6] achieved the accuracy of 72% with 4 classification words. [7] The accuracy calculated includes the tweets with stop words. Most of the stop words does not contribute any meaning to the sentences, since it is always suggested to remove the stop words [8] before sentiment analysis, this system also includes the character frequency count but it would have been more convenient if the work discussed regarding the contribution of character frequency in sentiment analysis. The overall accuracy calculated is on small data set. In [8] and [9] the algorithms are experimented on translated English reviews with 100 movie reviews in [8] and the dataset consists of 182 positive and 105 negative reviews in [9]. The research work [10] would have been more favorable if the dataset size is discussed. [11] The proposed work consists the dataset of all three languages together and the accuracy calculated is also considered with all three languages together. [12] The dataset with which the model is experimented is primary and adequate. As per author the limitation is the length of text. [13] The sentiment analysis is dominated by English language as it is the rich resource language over the Kannada language.

Drawbacks of existing works:

I. The existing systems which had achieved better accuracy are experimented with small data set.
II. In most of the work the processing of language is carried on English translated language instead of Indian regional language.
III. The sentiment is predicted by considering first 15 words only from each sentence and deleting the rest of words. This makes sentence to lose the actual meaning and in turn results in incorrect sentiment analysis.
IV. The system judges the sentiment of a sentence which includes the stop words. The sentiment of such sentences may be incorrect.
V. The system is developed with 3 to 4 Regional language classification words for sentiment analysis.

2 Methodology

Figure 1 shows the procedural steps of Sentiment Analysis of Twitter Feeds in Kannada. From twitter the reviews will be extracted. After fetching review from twitter, it will be tokenized i.e., review will be separated in smaller words. Data cleaning is the process of removing unwanted information from the reviews that doesn't give any meaning to the review such as punctuation, comma, etc. Stop words are the words in languages that do not give any meaning to the sentence such as the, which, and, etc. The process of breaking a word and extracting the root word is known as stemming. Classification is also called Text Tagging which is the process of organizing groups of text into its categories. Last step is calculation, where the polarity of review will be calculated.

2.1 Data collection

An activity of identifying, calculating and collecting the details on targeted data is known as data collection. It enables to answer useful questions and to calculate approximate outcomes. Data collection is useful component in many fields such as physical and social sciences, also in mankind and occupation. In data collection it is important to define and assemble the data in a proper way to maintain the honesty of investigation.

2.2 Tokenization

Tokenization is an action of dividing a sentence, paragraph or text document into small parts called tokens. Tokens involves characters, terms, phrases or words. For example, let us consider an example: "It is a pen". Creating a token by considering the space is the basic way of tokenization. After undergoing tokenization process the above sentence is reduced to tokens– "It", "is", "a", “pen”. Here each reduced tokens are words. We can perform tokenization on documents and sentences.

2.3 Data cleaning

Data cleaning is a salient step in NLP. Without cleaning of data, the dataset is like a collection of words which will not be understood by the computer. This step involves identifying

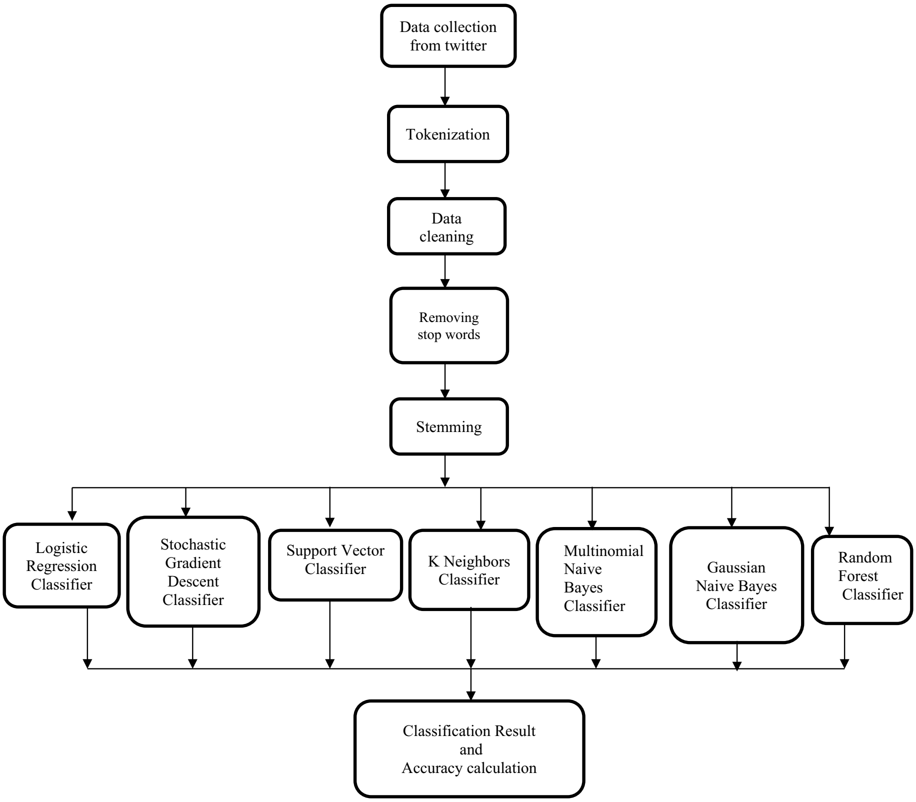

Data collection from twitter
Tokenization
Data cleaning
Removing stop words
Stemming
Logistic Regression Classifier
Stochastic Gradient Descent Classifier
Support Vector Classifier
K Neighbors Classifier
Multinomial Naive Bayes Classifier
Gaussian Naive Bayes Classifier
Random Forest Classifier
Classification Result and Accuracy calculation

Fig. 1 Steps involved in Sentiment Analysis of Twitter Feeds in Kannada. Tokenization is the breaking of sentences to words and stemming is to extract the root word which helps to classify the sentences

duplicate, erroneous, peripheral parts of the data and then modifying, replacing or deleting the unwanted data. In NLP, data cleaning involves removing various punctuation marks which include comma ‘,’, colon ‘:’, exclamation mark ‘!’, hyphen ‘-’, question mark ‘?’, apostrophe ‘’’, dash ‘-’, brackets ‘{}, [], ()’, semicolon ‘;’, quotation marks ‘ “”’, ellipsis (***) or (…).

2.4 Removing stop words

The words or terms that do not contribute any sense of weightage to the sentence in any language are called stop words. Removing of these stop words will not affect the actual meaning of sentence. Removing of these stop words will leads to decrease in the data size and time to train the model, with increase in performance and accuracy. In Natural Language Processing the NLTK library is one of the python libraries which is the oldest and most commonly used. NLTK library facilitates wide range of modules to support NLP process, corpus module is one among them which consists of list of stop words and helps to use these words to be excluded from input text and it also facilitates to extend this list if required by developer.

2.5 Stemming

The technology of breaking a locution to extract the base word from it is called stemming. Consider the example of words “sleeping” and “eating” are reduced to “sleep” and “eat” using stemming algorithm.

2.6 Classification

Classification process involves classifying the things into different groups based on their features or characters. Before Classification we use TF-IDF technique for feature extraction.

This work has been experimented with various classification algorithms such as Logistic Regression which is a supervised method that classifies based on probability of a word to be predicted, SGD classifier which establish a plain SGD learning routine supporting different loss function and penalty for classification, K-Neighbors classifier which is also a supervised learning method that makes the comparison between the new data and the available data and identifies the resemblance between them, later categorizes the data to the nearest matched class. Multinomial Naïve Bayes is also tested which works by using selective learning method. Gaussian Naïve Bayes classifier supports continuous data and follow gaussian normal distribution.

2.6.1 TF-IDF

To extract feature, TF-IDF (Term Frequency Inverse Document Frequency) [8] technique has been applied.

TF-IDF is computed by using two components:

Term Frequency (TF) is computed by calculating number of repetitions of each word with respect to sum of all of words from a document.
Inverse Document Frequency (IDF), computed by considering the sum of all documents in a database with respect to count of documents consisting of specific word.

After TF-IDF feature extraction, a sparse matrix is formed. This matrix is used for classification. We have used In-language classification to train the machine. The method is based on training the classifiers on the language for which analysis has to be done, and for this it is required to have an enough resource of the language. Thus, all training data and testing data are in the form of text in Kannada language. We used a variety of classifiers to train and test the data i.e., Linear SVC [1], Logistic Regression [7], SGD Classifier, SVC [7], K Neighbors Classifier, Multinomial NB [7], Gaussian NB and Random Forest Classifier [15].

3 Results and discussions

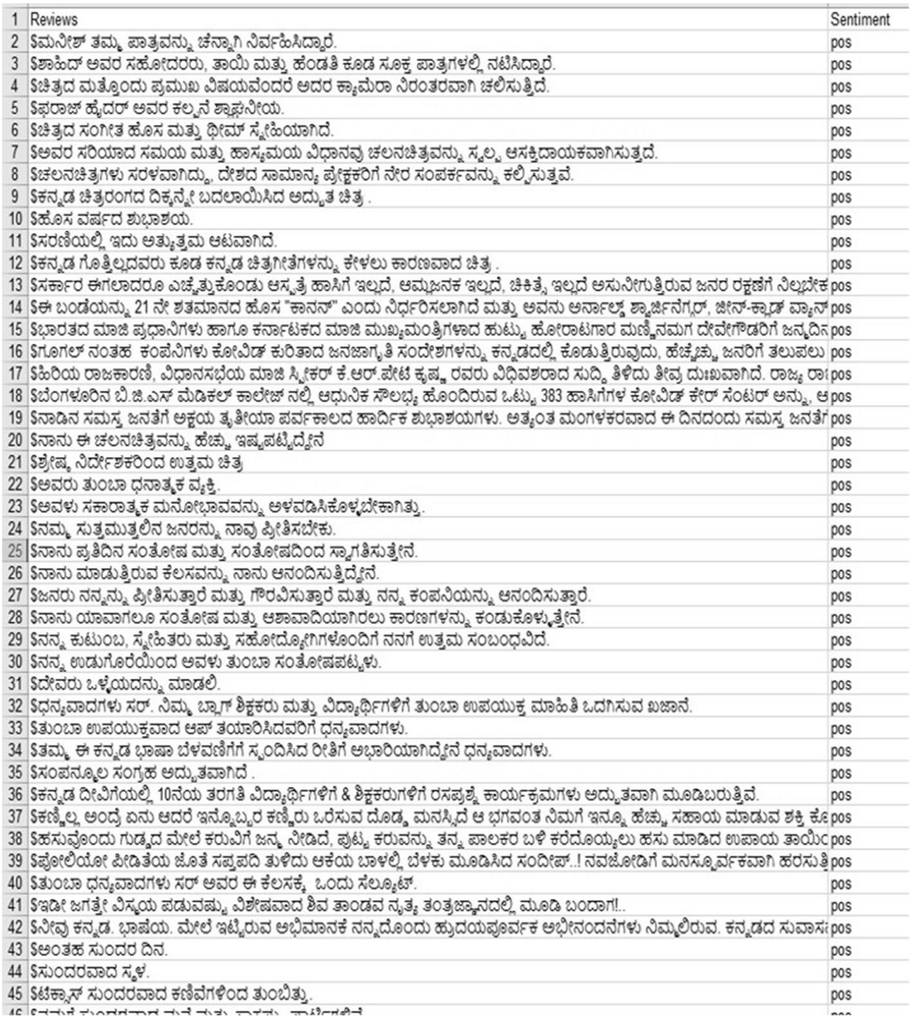
Figure 2 shows the data that has been collected from twitter feeds for sentiment analysis. Data set is prepared by collecting 1000 tweets. Tweets were divided into two classes namely positive and negative labelled respectively. These are stored as data in Comma-separated values (CSV) format.

The following figures illustrates the outcome of each intermediate steps of classification.

Figure 3 shows an example of tweets before applying tokenization. Figure 4 shows tweets after application of tokenization process. In tokenization the sentences are broken into certain tokens or words. These tokens are taken as input for data cleaning process. For example, the sentence "ಮನೀಶ್ ತಮ್ಮ ಪಾತ್ರವನ್ನು ಚೆನ್ನಾಗಿ ನಿರ್ವಹಿಸಿದ್ದಾರೆ.". This sentence is broken down into tokens like "ಮನೀಶ್", "ತಮ್ಮ", "ಪಾತ್ರವನ್ನು", "ಚೆನ್ನಾಗಿ", "ನಿರ್ವಹಿಸಿದ್ದಾರೆ", ".".

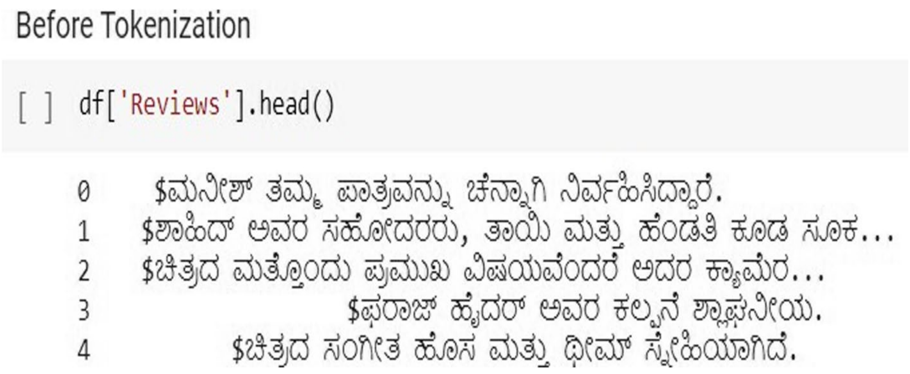
Figure 5 shows an example of tweet before applying cleaning process. Figure 6 shows an example after applying data cleaning process. Data cleaning process involves removing of unwanted data such as punctuation marks which do not help for sentiment analysis. In the above example punctuation marks like comma, full stop and dollar are removed.

Figure 7 shows examples of tweets before removing stop words and Fig. 8 shows examples of Kannada tweets after eliminating stop words. The words that do not contribute any meaning to the sentence are categorized as stop words. Due to the removal of stop words,



1	Reviews	Sentiment
2	$ಮನೀಶ್ ತಮ್ಮ ಪಾತ್ರವನ್ನು ಚೆನ್ನಾಗಿ ನಿರ್ವಹಿಸಿದ್ದಾರೆ.	pos
3	$ಶಾಹಿದ್ ಅವರ ಸಹೋದರರು, ತಾಯಿ ಮತ್ತು ಹೆಂಡತಿ ಕೂಡ ಸೂಕ್ತ ಪಾತ್ರಗಳಲ್ಲಿ ನಟಿಸಿದ್ದಾರೆ.	pos
4	$ಚಿತ್ರದ ಮತ್ತೊಂದು ಪ್ರಮುಖ ವಿಷಯವೆಂದರೆ ಅದರ ಕ್ಯಾಮೆರಾ ನಿರಂತರವಾಗಿ ಚಲಿಸುತ್ತಿದೆ.	pos
5	$ಫರಾಜ್ ಹೈದರ್ ಅವರ ಕಲ್ಪನೆ ಶ್ಲಾಘನೀಯ.	pos
6	$ಚಿತ್ರದ ಸಂಗೀತ ಹೊಸ ಮತ್ತು ಥೀಮ್ ಸ್ನೇಹಿಯಾಗಿದೆ.	pos
7	$ಅವರ ಸರಿಯಾದ ಸಮಯ ಮತ್ತು ಹಾಸ್ಯಮಯ ವಿಧಾನವು ಚಲನಚಿತ್ರವನ್ನು ಸ್ವಲ್ಪ ಆಸಕ್ತಿದಾಯಕವಾಗಿಸುತ್ತದೆ.	pos
8	$ಚಲನಚಿತ್ರಗಳು ಸರಳವಾಗಿದ್ದು, ದೇಶದ ಸಾಮಾನ್ಯ ಪ್ರೇಕ್ಷಕರಿಗೆ ನೇರ ಸಂಪರ್ಕವನ್ನು ಕಲ್ಪಿಸುತ್ತವೆ.	pos
9	$ಕನ್ನಡ ಚಿತ್ರರಂಗದ ದಿಕ್ಕನ್ನೇ ಬದಲಾಯಿಸಿದ ಅದ್ಭುತ ಚಿತ್ರ .	pos
10	$ಹೊಸ ವರ್ಷದ ಶುಭಾಶಯ.	pos
11	$ಸರಣಿಯಲ್ಲಿ ಇದು ಅತ್ಯುತ್ತಮ ಆಟವಾಗಿದೆ.	pos
12	$ಕನ್ನಡ ಗೊತ್ತಿಲ್ಲದವರು ಕೂಡ ಕನ್ನಡ ಚಿತ್ರಗೀತೆಗಳನ್ನು ಕೇಳಲು ಕಾರಣವಾದ ಚಿತ್ರ .	pos
13	$ಸರ್ಕಾರ ಈಗಲಾದರೂ ಎಚ್ಚೆತ್ತುಕೊಂಡು ಆಸ್ಪತ್ರೆ ಹಾಸಿಗೆ ಇಲ್ಲದೆ, ಆಮ್ಲಜನಕ ಇಲ್ಲದೆ, ಚಿಕಿತ್ಸೆ ಇಲ್ಲದೆ ಅಸುನೀಗುತ್ತಿರುವ ಜನರ ರಕ್ಷಣೆಗೆ ನಿಲ್ಲಬೇಕ	pos
14	$ಈ ಬಂಡೆಯನ್ನು 21 ನೇ ಶತಮಾನದ ಹೊಸ "ಕಾನನ್" ಎಂದು ನಿರ್ಧರಿಸಲಾಗಿದೆ ಮತ್ತು ಅವನು ಅರ್ನಾಲ್ಡ್ ಶ್ವಾರ್ಜಿನೆಗ್ಗರ್, ಜೀನ್-ಕ್ಲಾಡ್ ವ್ಯಾನ್	pos
15	$ಭಾರತದ ಮಾಜಿ ಪ್ರಧಾನಿಗಳು ಹಾಗೂ ಕರ್ನಾಟಕದ ಮಾಜಿ ಮುಖ್ಯಮಂತ್ರಿಗಳಾದ ಹುಟ್ಟು ಹೋರಾಟಗಾರ ಮಣ್ಣಿನಮಗ ದೇವೇಗೌಡರಿಗೆ ಜನ್ಮದಿನ	pos
16	$ಗೂಗಲ್ ನಂತಹ ಕಂಪೆನಿಗಳು ಕೋವಿಡ್ ಕುರಿತಾದ ಜನಜಾಗೃತಿ ಸಂದೇಶಗಳನ್ನು ಕನ್ನಡದಲ್ಲಿ ಕೊಡುತ್ತಿರುವುದು, ಹೆಚ್ಚೆಚ್ಚು ಜನರಿಗೆ ತಲುಪಲು	pos
17	$ಹಿರಿಯ ರಾಜಕಾರಣಿ, ವಿಧಾನಸಭೆಯ ಮಾಜಿ ಸ್ಪೀಕರ್ ಕೆ.ಆರ್.ಪೇಟೆ ಕೃಷ್ಣ ರವರು ವಿಧಿವಶರಾದ ಸುದ್ದಿ ತಿಳಿದು ತೀವ್ರ ದುಃಖವಾಗಿದೆ. ರಾಜ್ಯ ರಾ	pos
18	$ಬೆಂಗಳೂರಿನ ಬಿ.ಜಿ.ಎಸ್ ಮೆಡಿಕಲ್ ಕಾಲೇಜ್ ನಲ್ಲಿ ಆಧುನಿಕ ಸೌಲಭ್ಯ ಹೊಂದಿರುವ ಒಟ್ಟು 383 ಹಾಸಿಗೆಗಳ ಕೋವಿಡ್ ಕೇರ್ ಸೆಂಟರ್ ಅನ್ನು, ಆ	pos
19	$ನಾಡಿನ ಸಮಸ್ತ ಜನತೆಗೆ ಅಕ್ಷಯ ತೃತೀಯಾ ಪರ್ವಕಾಲದ ಹಾರ್ದಿಕ ಶುಭಾಶಯಗಳು. ಅತ್ಯಂತ ಮಂಗಳಕರವಾದ ಈ ದಿನದಂದು ಸಮಸ್ತ ಜನತೆ	pos
20	$ನಾನು ಈ ಚಲನಚಿತ್ರವನ್ನು ಹೆಚ್ಚು ಇಷ್ಟಪಟ್ಟಿದ್ದೇನೆ	pos
21	$ಶ್ರೇಷ್ಠ ನಿರ್ದೇಶಕರಿಂದ ಉತ್ತಮ ಚಿತ್ರ	pos
22	$ಅವರು ತುಂಬಾ ಧನಾತ್ಮಕ ವ್ಯಕ್ತಿ.	pos
23	$ಅವಳು ಸಕಾರಾತ್ಮಕ ಮನೋಭಾವವನ್ನು ಅಳವಡಿಸಿಕೊಳ್ಳಬೇಕಾಗಿತ್ತು.	pos
24	$ನಮ್ಮ ಸುತ್ತಮುತ್ತಲಿನ ಜನರನ್ನು ನಾವು ಪ್ರೀತಿಸಬೇಕು.	pos
25	$ನಾನು ಪ್ರತಿದಿನ ಸಂತೋಷ ಮತ್ತು ಸಂತೋಷದಿಂದ ಸ್ವಾಗತಿಸುತ್ತೇನೆ.	pos
26	$ನಾನು ಮಾಡುತ್ತಿರುವ ಕೆಲಸವನ್ನು ನಾನು ಆನಂದಿಸುತ್ತಿದ್ದೇನೆ.	pos
27	$ಜನರು ನನ್ನನ್ನು ಪ್ರೀತಿಸುತ್ತಾರೆ ಮತ್ತು ಗೌರವಿಸುತ್ತಾರೆ ಮತ್ತು ನನ್ನ ಕಂಪನಿಯನ್ನು ಆನಂದಿಸುತ್ತಾರೆ.	pos
28	$ನಾನು ಯಾವಾಗಲೂ ಸಂತೋಷ ಮತ್ತು ಆಶಾವಾದಿಯಾಗಿರಲು ಕಾರಣಗಳನ್ನು ಕಂಡುಕೊಳ್ಳುತ್ತೇನೆ.	pos
29	$ನನ್ನ ಕುಟುಂಬ, ಸ್ನೇಹಿತರು ಮತ್ತು ಸಹೋದ್ಯೋಗಿಗಳೊಂದಿಗೆ ನನಗೆ ಉತ್ತಮ ಸಂಬಂಧವಿದೆ.	pos
30	$ನನ್ನ ಉಡುಗೊರೆಯಿಂದ ಅವಳು ತುಂಬಾ ಸಂತೋಷಪಟ್ಟಳು.	pos
31	$ದೇವರು ಒಳ್ಳೆಯದನ್ನು ಮಾಡಲಿ.	pos
32	$ಧನ್ಯವಾದಗಳು ಸರ್. ನಿಮ್ಮ ಬ್ಲಾಗ್ ಶಿಕ್ಷಕರು ಮತ್ತು ವಿದ್ಯಾರ್ಥಿಗಳಿಗೆ ತುಂಬಾ ಉಪಯುಕ್ತ ಮಾಹಿತಿ ಒದಗಿಸುವ ಖಜಾನೆ.	pos
33	$ತುಂಬಾ ಉಪಯುಕ್ತವಾದ ಆಪ್ ತಯಾರಿಸಿದವರಿಗೆ ಧನ್ಯವಾದಗಳು.	pos
34	$ತಮ್ಮ ಈ ಕನ್ನಡ ಭಾಷಾ ಬೆಳವಣಿಗೆಗೆ ಸ್ಪಂದಿಸಿದ ರೀತಿಗೆ ಅಭಾರಿಯಾಗಿದ್ದೇನೆ ಧನ್ಯವಾದಗಳು.	pos
35	$ಸಂಪನ್ಮೂಲ ಸಂಗ್ರಹ ಅದ್ಭುತವಾಗಿದೆ .	pos
36	$ಕನ್ನಡ ದೀವಿಗೆಯಲ್ಲಿ 10ನೆಯ ತರಗತಿ ವಿದ್ಯಾರ್ಥಿಗಳಿಗೆ & ಶಿಕ್ಷಕರುಗಳಿಗೆ ರಸಪ್ರಶ್ನೆ ಕಾರ್ಯಕ್ರಮಗಳು ಅದ್ಭುತವಾಗಿ ಮೂಡಿಬರುತ್ತಿವೆ.	pos
37	$ಕಣ್ಣಿಲ್ಲ ಅಂದ್ರೆ ಏನು ಆದರೆ ಇನ್ನೊಬ್ಬರ ಕಣ್ಣೀರು ಒರೆಸುವ ದೊಡ್ಡ ಮನಸ್ಸಿದೆ ಆ ಭಗವಂತ ನಿಮಗೆ ಇನ್ನೂ ಹೆಚ್ಚು ಸಹಾಯ ಮಾಡುವ ಶಕ್ತಿ ಕೊ	pos
38	$ಹಸುವೊಂದು ಗುಡ್ಡದ ಮೇಲೆ ಕರುವಿಗೆ ಜನ್ಮ ನೀಡಿದೆ, ಪುಟ್ಟ ಕರುವನ್ನು ತನ್ನ ಪಾಲಕರ ಬಳಿ ಕರೆದೊಯ್ಯಲು ಹಸು ಮಾಡಿದ ಉಪಾಯ ತಾಯಿಂ	pos
39	$ಪೋಲಿಯೋ ಪೀಡಿತೆಯ ಜೊತೆ ಸಪ್ತಪದಿ ತುಳಿದು ಆಕೆಯ ಬಾಳಲ್ಲಿ ಬೆಳಕು ಮೂಡಿಸಿದ ಸಂದೀಪ್..! ನವಜೋಡಿಗೆ ಮನಸ್ಪೂರ್ವಕವಾಗಿ ಹರಸುತ್ತಿ	pos
40	$ತುಂಬಾ ಧನ್ಯವಾದಗಳು ಸರ್ ಅವರ ಈ ಕೆಲಸಕ್ಕೆ ಒಂದು ಸೆಲ್ಯೂಟ್.	pos
41	$ಇಡೀ ಜಗತ್ತೇ ವಿಸ್ಮಯ ಪಡುವಷ್ಟು ವಿಶೇಷವಾದ ಶಿವ ತಾಂಡವ ನೃತ್ಯ ತಂತ್ರಜ್ಞಾನದಲ್ಲಿ ಮೂಡಿ ಬಂದಾಗ!..	pos
42	$ನೀವು ಕನ್ನಡ. ಭಾಷೆಯ. ಮೇಲೆ ಇಟ್ಟಿರುವ ಅಭಿಮಾನಕೆ ನನ್ನದೊಂದು ಹೃದಯಪೂರ್ವಕ ಅಭಿನಂದನೆಗಳು ನಿಮ್ಮಲಿರುವ. ಕನ್ನಡದ ಸುವಾಸ	pos
43	$ಅಂತಹ ಸುಂದರ ದಿನ.	pos
44	$ಸುಂದರವಾದ ಸ್ಥಳ.	pos
45	$ಟೆಕ್ಸಾಸ್ ಸುಂದರವಾದ ಕಣಿವೆಗಳಿಂದ ತುಂಬಿತ್ತು.	pos
46	[illegible]	[illegible]

Fig. 2 Sample data from twitter. Few Kannada tweets from data collected from twitter

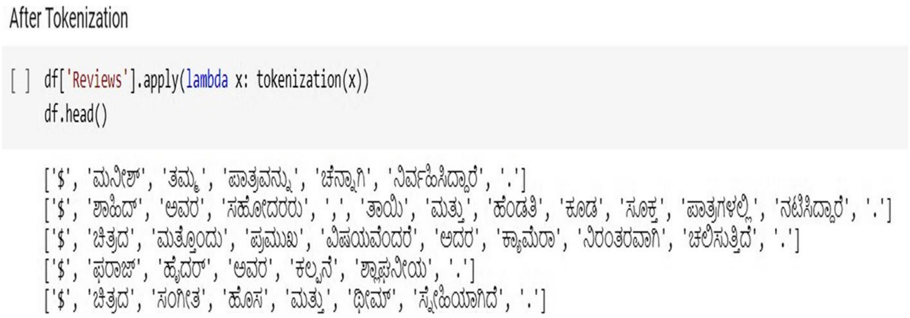


Before Tokenization

```
[ ]  df['Reviews'].head()
```

```
0     $ಮನೀಶ್ ತಮ್ಮ ಪಾತ್ರವನ್ನು ಚೆನ್ನಾಗಿ ನಿರ್ವಹಿಸಿದ್ದಾರೆ.
1    $ಶಾಹಿದ್ ಅವರ ಸಹೋದರರು, ತಾಯಿ ಮತ್ತು ಹೆಂಡತಿ ಕೂಡ ಸೂಕ...
2    $ಚಿತ್ರದ ಮತ್ತೊಂದು ಪ್ರಮುಖ ವಿಷಯವೆಂದರೆ ಅದರ ಕ್ಯಾಮೆರ...
3                   $ಫರಾಜ್ ಹೈದರ್ ಅವರ ಕಲ್ಪನೆ ಶ್ಲಾಘನೀಯ.
4          $ಚಿತ್ರದ ಸಂಗೀತ ಹೊಸ ಮತ್ತು ಥೀಮ್ ಸ್ನೇಹಿಯಾಗಿದೆ.
```

Fig. 3 Screenshot of Kannada tweets before processing tokenization



After Tokenization

```
[ ] df['Reviews'].apply(lambda x: tokenization(x))
    df.head()
```

```
['$', 'ಮನೀಶ್', 'ತಮ್ಮ', 'ಪಾತ್ರವನ್ನು', 'ಚೆನ್ನಾಗಿ', 'ನಿರ್ವಹಿಸಿದ್ದಾರೆ', '.']
['$', 'ಶಾಹಿದ್', 'ಅವರ', 'ಸಹೋದರರು', ',', 'ತಾಯಿ', 'ಮತ್ತು', 'ಹೆಂಡತಿ', 'ಕೂಡ', 'ಸೂಕ್ತ', 'ಪಾತ್ರಗಳಲ್ಲಿ', 'ನಟಿಸಿದ್ದಾರೆ', '.']
['$', 'ಚಿತ್ರದ', 'ಮತ್ತೊಂದು', 'ಪ್ರಮುಖ', 'ವಿಷಯವೆಂದರೆ', 'ಅದರ', 'ಕ್ಯಾಮೆರಾ', 'ನಿರಂತರವಾಗಿ', 'ಚಲಿಸುತ್ತಿದೆ', '.']
['$', 'ಫರಾಜ್', 'ಹೈದರ್', 'ಅವರ', 'ಕಲ್ಪನೆ', 'ಶ್ಲಾಘನೀಯ', '.']
['$', 'ಚಿತ್ರದ', 'ಸಂಗೀತ', 'ಹೊಸ', 'ಮತ್ತು', 'ಥೀಮ್', 'ಸ್ನೇಹಿಯಾಗಿದೆ', '.']
```

Fig. 4 Screenshot of Kannada tweets after processing tokenization

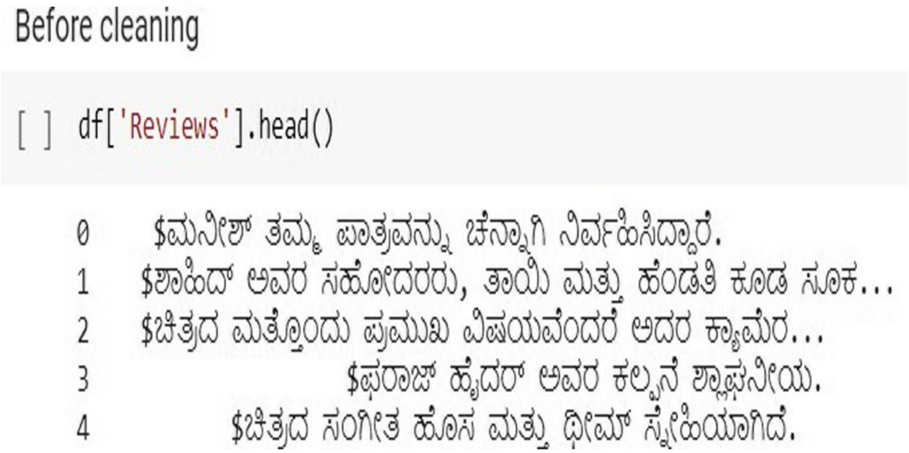
the dataset size is reduced in turn which decreases the training period without affecting the accuracy of a system.

Figure 9 shows the sentences before stemming and Fig. 10 shows an example after application of stemming process. The technique of minimizing a word to its base word by excluding the suffix of a word is called stemming. In this above example in first sentence the word "ಪಾತ್ರವನ್ನು" is reduced to "ಪಾತ್ರ".

Figure 11 shows an example of Custom Input where users can give sentences manually in order to classify it as positive or negative. Figure 12 shows an example of tweet after application of cleaning and stemming process. Figure 13 shows an example after applying classification method. This classification method will classify a tweet as positive or negative tweet. If it is a positive sentence it will print "It is a positive sentence" else it will print "It is a negative sentence".

3.1 Comparitive analysis of algorithms

As discussed in earlier sections, the tweet will be preprocessed before classification. Classification is the very important step in Sentiment Analysis in which tweets are classified into either of two classes i.e., positive or negative. In order to classify tweets, we use classification algorithms. In our model, we have used Logistic Regression, SGD

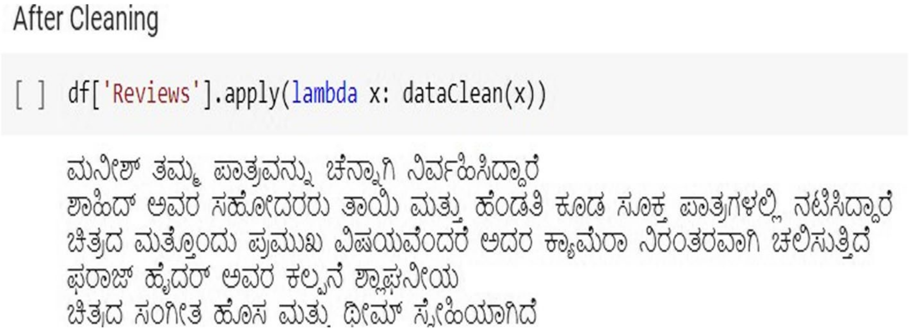


Before cleaning

```
[ ] df['Reviews'].head()
```

```
0     $ಮನೀಶ್ ತಮ್ಮ ಪಾತ್ರವನ್ನು ಚೆನ್ನಾಗಿ ನಿರ್ವಹಿಸಿದ್ದಾರೆ.
1    $ಶಾಹಿದ್ ಅವರ ಸಹೋದರರು, ತಾಯಿ ಮತ್ತು ಹೆಂಡತಿ ಕೂಡ ಸೂಕ...
2    $ಚಿತ್ರದ ಮತ್ತೊಂದು ಪ್ರಮುಖ ವಿಷಯವೆಂದರೆ ಅದರ ಕ್ಯಾಮೆರ...
3                   $ಫರಾಜ್ ಹೈದರ್ ಅವರ ಕಲ್ಪನೆ ಶ್ಲಾಘನೀಯ.
4           $ಚಿತ್ರದ ಸಂಗೀತ ಹೊಸ ಮತ್ತು ಥೀಮ್ ಸ್ನೇಹಿಯಾಗಿದೆ.
```

Fig. 5 Screenshot of Kannada tweets before processing cleaning step

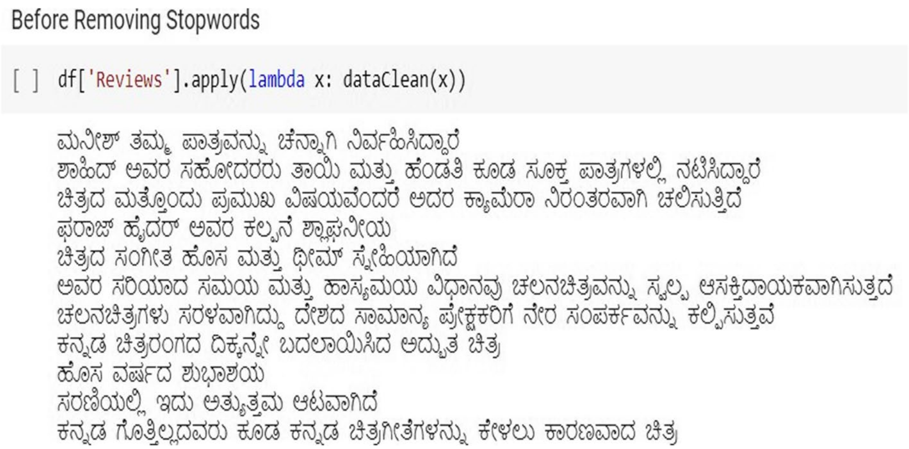


After Cleaning

```
[ ] df['Reviews'].apply(lambda x: dataClean(x))
```

```
ಮನೀಶ್ ತಮ್ಮ ಪಾತ್ರವನ್ನು ಚೆನ್ನಾಗಿ ನಿರ್ವಹಿಸಿದ್ದಾರೆ
ಶಾಹಿದ್ ಅವರ ಸಹೋದರರು ತಾಯಿ ಮತ್ತು ಹೆಂಡತಿ ಕೂಡ ಸೂಕ್ತ ಪಾತ್ರಗಳಲ್ಲಿ ನಟಿಸಿದ್ದಾರೆ
ಚಿತ್ರದ ಮತ್ತೊಂದು ಪ್ರಮುಖ ವಿಷಯವೆಂದರೆ ಅದರ ಕ್ಯಾಮೆರಾ ನಿರಂತರವಾಗಿ ಚಲಿಸುತ್ತಿದೆ
ಫರಾಜ್ ಹೈದರ್ ಅವರ ಕಲ್ಪನೆ ಶ್ಲಾಘನೀಯ
ಚಿತ್ರದ ಸಂಗೀತ ಹೊಸ ಮತ್ತು ಥೀಮ್ ಸ್ನೇಹಿಯಾಗಿದೆ
```

Fig. 6 Screenshot of Kannada tweets after processing cleaning step. Cleaning process involves removal of punctuation marks

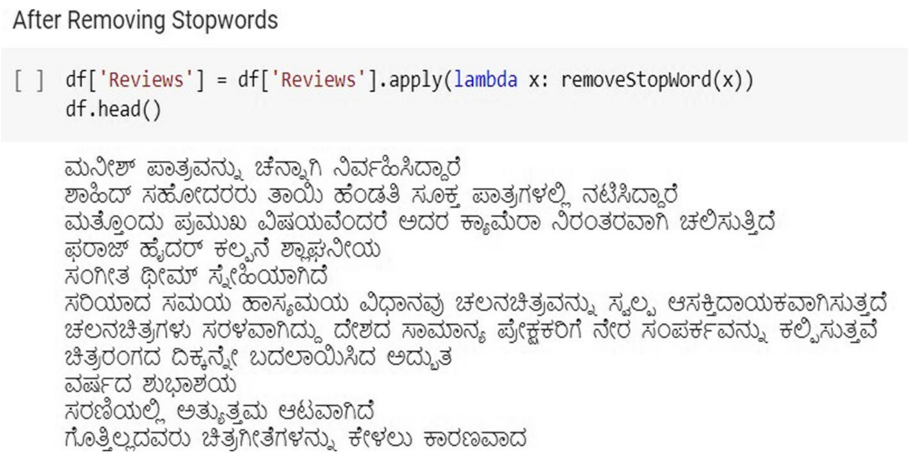


Before Removing Stopwords

```
[ ] df['Reviews'].apply(lambda x: dataClean(x))
```

```
ಮನೀಶ್ ತಮ್ಮ ಪಾತ್ರವನ್ನು ಚೆನ್ನಾಗಿ ನಿರ್ವಹಿಸಿದ್ದಾರೆ
ಶಾಹಿದ್ ಅವರ ಸಹೋದರರು ತಾಯಿ ಮತ್ತು ಹೆಂಡತಿ ಕೂಡ ಸೂಕ್ತ ಪಾತ್ರಗಳಲ್ಲಿ ನಟಿಸಿದ್ದಾರೆ
ಚಿತ್ರದ ಮತ್ತೊಂದು ಪ್ರಮುಖ ವಿಷಯವೆಂದರೆ ಅದರ ಕ್ಯಾಮೆರಾ ನಿರಂತರವಾಗಿ ಚಲಿಸುತ್ತಿದೆ
ಫರಾಜ್ ಹೈದರ್ ಅವರ ಕಲ್ಪನೆ ಶ್ಲಾಘನೀಯ
ಚಿತ್ರದ ಸಂಗೀತ ಹೊಸ ಮತ್ತು ಥೀಮ್ ಸ್ನೇಹಿಯಾಗಿದೆ
ಅವರ ಸರಿಯಾದ ಸಮಯ ಮತ್ತು ಹಾಸ್ಯಮಯ ವಿಧಾನವು ಚಲನಚಿತ್ರವನ್ನು ಸ್ವಲ್ಪ ಆಸಕ್ತಿದಾಯಕವಾಗಿಸುತ್ತದೆ
ಚಲನಚಿತ್ರಗಳು ಸರಳವಾಗಿದ್ದು ದೇಶದ ಸಾಮಾನ್ಯ ಪ್ರೇಕ್ಷಕರಿಗೆ ನೇರ ಸಂಪರ್ಕವನ್ನು ಕಲ್ಪಿಸುತ್ತವೆ
ಕನ್ನಡ ಚಿತ್ರರಂಗದ ದಿಕ್ಕನ್ನೇ ಬದಲಾಯಿಸಿದ ಅದ್ಭುತ ಚಿತ್ರ
ಹೊಸ ವರ್ಷದ ಶುಭಾಶಯ
ಸರಣಿಯಲ್ಲಿ ಇದು ಅತ್ಯುತ್ತಮ ಆಟವಾಗಿದೆ
ಕನ್ನಡ ಗೊತ್ತಿಲ್ಲದವರು ಕೂಡ ಕನ್ನಡ ಚಿತ್ರಗೀತೆಗಳನ್ನು ಕೇಳಲು ಕಾರಣವಾದ ಚಿತ್ರ
```

Fig. 7 Screenshot of Kannada tweets before removing stop words

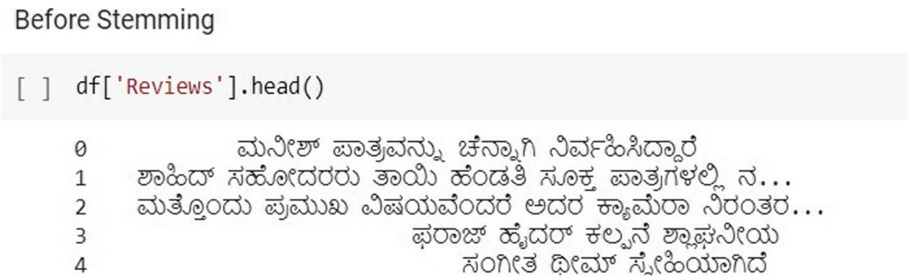


After Removing Stopwords

```
[ ]  df['Reviews'] = df['Reviews'].apply(lambda x: removeStopWord(x))
     df.head()
```

```
ಮನೀಶ್ ಪಾತ್ರವನ್ನು ಚೆನ್ನಾಗಿ ನಿರ್ವಹಿಸಿದ್ದಾರೆ
ಶಾಹಿದ್ ಸಹೋದರರು ತಾಯಿ ಹೆಂಡತಿ ಸೂಕ್ತ ಪಾತ್ರಗಳಲ್ಲಿ ನಟಿಸಿದ್ದಾರೆ
ಮತ್ತೊಂದು ಪ್ರಮುಖ ವಿಷಯವೆಂದರೆ ಅದರ ಕ್ಯಾಮೆರಾ ನಿರಂತರವಾಗಿ ಚಲಿಸುತ್ತಿದೆ
ಫರಾಜ್ ಹೈದರ್ ಕಲ್ಪನೆ ಶ್ಲಾಘನೀಯ
ಸಂಗೀತ ಥೀಮ್ ಸ್ನೇಹಿಯಾಗಿದೆ
ಸರಿಯಾದ ಸಮಯ ಹಾಸ್ಯಮಯ ವಿಧಾನವು ಚಲನಚಿತ್ರವನ್ನು ಸ್ವಲ್ಪ ಆಸಕ್ತಿದಾಯಕವಾಗಿಸುತ್ತದೆ
ಚಲನಚಿತ್ರಗಳು ಸರಳವಾಗಿದ್ದು ದೇಶದ ಸಾಮಾನ್ಯ ಪ್ರೇಕ್ಷಕರಿಗೆ ನೇರ ಸಂಪರ್ಕವನ್ನು ಕಲ್ಪಿಸುತ್ತವೆ
ಚಿತ್ರರಂಗದ ದಿಕ್ಕನ್ನೇ ಬದಲಾಯಿಸಿದ ಅದ್ಭುತ
ವರ್ಷದ ಶುಭಾಶಯ
ಸರಣಿಯಲ್ಲಿ ಅತ್ಯುತ್ತಮ ಆಟವಾಗಿದೆ
ಗೊತ್ತಿಲ್ಲದವರು ಚಿತ್ರಗೀತೆಗಳನ್ನು ಕೇಳಲು ಕಾರಣವಾದ
```

Fig. 8 Screenshot of Kannada tweets after removing stop words. Stop words are the conjunctions in any language

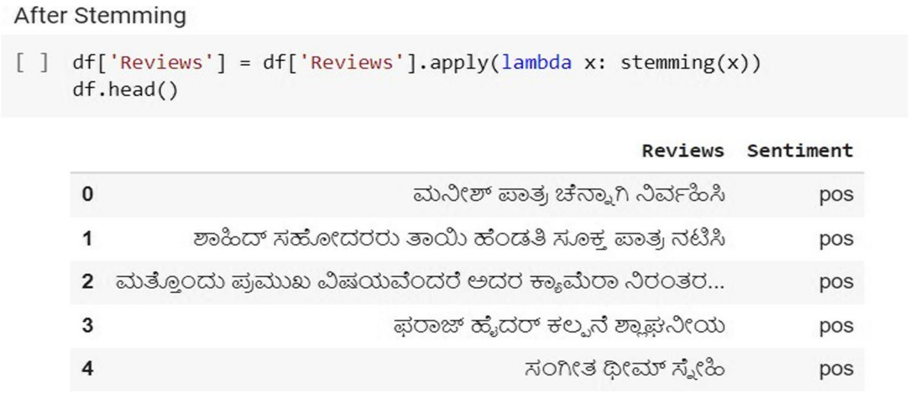


Before Stemming

```
[ ]  df['Reviews'].head()
```

```
0              ಮನೀಶ್ ಪಾತ್ರವನ್ನು ಚೆನ್ನಾಗಿ ನಿರ್ವಹಿಸಿದ್ದಾರೆ
1    ಶಾಹಿದ್ ಸಹೋದರರು ತಾಯಿ ಹೆಂಡತಿ ಸೂಕ್ತ ಪಾತ್ರಗಳಲ್ಲಿ ನ...
2    ಮತ್ತೊಂದು ಪ್ರಮುಖ ವಿಷಯವೆಂದರೆ ಅದರ ಕ್ಯಾಮೆರಾ ನಿರಂತರ...
3                          ಫರಾಜ್ ಹೈದರ್ ಕಲ್ಪನೆ ಶ್ಲಾಘನೀಯ
4                             ಸಂಗೀತ ಥೀಮ್ ಸ್ನೇಹಿಯಾಗಿದೆ
```

Fig. 9 Screenshot of Kannada tweets before applying stemming procedure

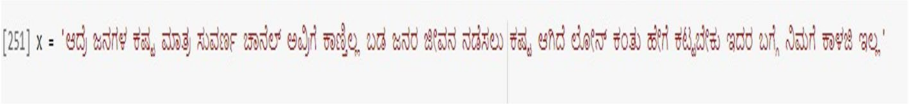


After Stemming

```
[ ]  df['Reviews'] = df['Reviews'].apply(lambda x: stemming(x))
     df.head()
```

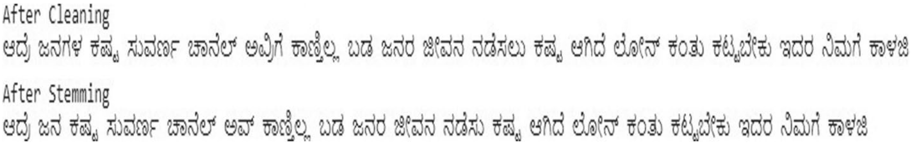
	Reviews	Sentiment
0	ಮನೀಶ್ ಪಾತ್ರ ಚೆನ್ನಾಗಿ ನಿರ್ವಹಿಸಿ	pos
1	ಶಾಹಿದ್ ಸಹೋದರರು ತಾಯಿ ಹೆಂಡತಿ ಸೂಕ್ತ ಪಾತ್ರ ನಟಿಸಿ	pos
2	ಮತ್ತೊಂದು ಪ್ರಮುಖ ವಿಷಯವೆಂದರೆ ಅದರ ಕ್ಯಾಮೆರಾ ನಿರಂತರ...	pos
3	ಫರಾಜ್ ಹೈದರ್ ಕಲ್ಪನೆ ಶ್ಲಾಘನೀಯ	pos
4	ಸಂಗೀತ ಥೀಮ್ ಸ್ನೇಹಿ	pos

Fig. 10 Screenshot of Kannada tweets after applying stemming procedure. Stemming is the procedure of deriving the root words



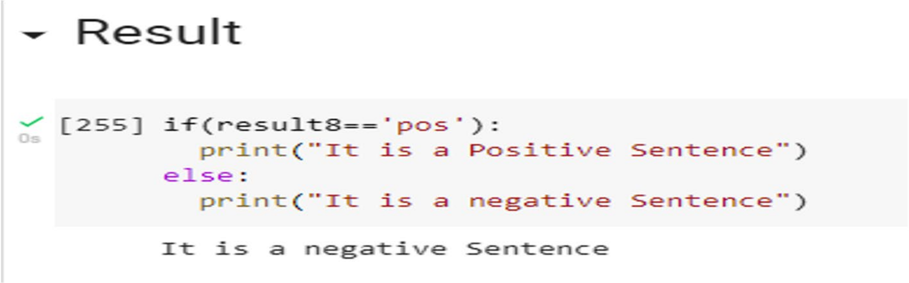
```
[251] x = 'ಆದ್ರೆ ಜನಗಳ ಕಷ್ಟ ಮಾತ್ರ ಸುವರ್ಣ ಚಾನೆಲ್ ಅವ್ರಿಗೆ ಕಾಣ್ತಿಲ್ಲ ಬಡ ಜನರ ಜೀವನ ನಡೆಸಲು ಕಷ್ಟ ಆಗಿದೆ ಲೋನ್ ಕಂತು ಹೇಗೆ ಕಟ್ಟ್ಬೇಕು ಇದರ ಬಗ್ಗೆ ನಿಮಗೆ ಕಾಳಜಿ ಇಲ್ಲ'
```

Fig. 11 Screenshot of one sample Kannada tweet considered for demonstration



```
After Cleaning
ಆದ್ರೆ ಜನಗಳ ಕಷ್ಟ ಸುವರ್ಣ ಚಾನೆಲ್ ಅವ್ರಿಗೆ ಕಾಣ್ತಿಲ್ಲ ಬಡ ಜನರ ಜೀವನ ನಡೆಸಲು ಕಷ್ಟ ಆಗಿದೆ ಲೋನ್ ಕಂತು ಕಟ್ಟ್ಬೇಕು ಇದರ ನಿಮಗೆ ಕಾಳಜಿ
After Stemming
ಆದ್ರೆ ಜನ ಕಷ್ಟ ಸುವರ್ಣ ಚಾನೆಲ್ ಅವ್ ಕಾಣ್ತಿಲ್ಲ ಬಡ ಜನರ ಜೀವನ ನಡೆಸು ಕಷ್ಟ ಆಗಿದೆ ಲೋನ್ ಕಂತು ಕಟ್ಟ್ಬೇಕು ಇದರ ನಿಮಗೆ ಕಾಳಜಿ
```

Fig. 12 Screenshot of result of cleaning and stemming procedures for sample tweet



```
[255] if(result8=='pos'):
        print("It is a Positive Sentence")
      else:
        print("It is a negative Sentence")

      It is a negative Sentence
```

Fig. 13 Screen shot of result of classification for sample tweet which categorizes it as negative

Classifier, SVC, K-Neighbors Classifier, Multinomial NB, Gaussian NB and Random Forest Classifier. The goal is to classify tweets using these algorithms and measure efficiency.

3.1.1 Performance measures

We have used the following performance measures to measure the efficiency of the various machine learning algorithms and found that proposed Multinominal Naïve Bayes algorithm has performed better as compared to other machine learning algorithms with the accuracy of 75%.

Precision Percentage of correct predictions. This parameter of algorithm depicts the number true predictions for true data. It is expressed with the proportion of true positive (TP) values and the addition of true and false positive (FP) values.

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP})$$

Recall Percentage of positive cases. This parameter of algorithm Recall depicts the positive instances. It is expressed with the proportion of true positive values and the addition of true and false negative (FN) values.

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN})$$

Table 1 Accuracy of classifiers

SL.NO	Classifier	Accuracy
1	Logistic Regression	69.44%
2	SGD Classifier	68.06%
3	SVC	72.22%
4	K Neighbors Classifiers	48.61%
5	Multinominal NB	75.0%
6	Gaussian NB	62.5%
7	Random Forest Classifier	65.28%

F1 score It represents correct positive prediction percentage. Mathematically F1 score is a valued harmonic mean in which predictions average ratio is calculated and best score 1.0 will be given to high data as compared to low data. Generally, F1 score is used to differentiate among classifier models and cannot be used for accuracy measurement.

Support The parameter which decides the count of each class within the dataset is called as support. If the training data has imbalance support, then it depicts the resulted scores of classifiers has the structural weakness and may require stratified sampling or rebalancing. There is no change in support among the different models, instead it does the interpretation on the evaluation process.

Accuracy True positive values are added to true negative values and result is divided by sample count from a document which leads to the accuracy. This calculation is justifiable if the model is balanced and not appropriate if there is a class imbalance.

Macro average Macro Average is the average of precisions of classes without considering the proportion.

$$\text{precision\_positive} = \text{X}$$
$$\text{precision\_negative} = \text{Y}$$
$$\text{Macro Average Precision} = (\text{X} + \text{Y})/2$$

Weighted average Weighted Average is the average considering the proportion. precision_positive = 0.35

$$\text{precision\_negative} = 0.24$$
$$\text{samples\_positive} = 39$$
$$\text{samples\_negative} = 27$$
$$\text{total\_samples} = 100 \ \text{proportion\_class\_0} = 39/100 = 0.39 \ \text{proportion\_class\_1} = 27/100 = 0.27$$
$$\text{Weighted Average Precision} = ((0.35 * 0.39) \ + (0.24 * 0.27)$$

3.1.2 Performance analysis of algorithms

Table 1 depicts the calculated accuracy of different algorithms over the dataset and the Fig. 14 shows the accuracy comparison of all the algorithms used in our model. Our model was trained with 1000 Kannada Sentences. Training and testing are two categories of datasets considered. Testing data was 0.2 of all data set.

From the comparison, we found that Multinomial NB performed better among all the classifiers. It had an accuracy score of 75.0%. SVC showed accuracy of 72.22% whereas Linear SVC showed 70.83% accuracy. Logistic Regression had an accuracy of 69.44%.

SGD Classifier showed accuracy of 68.06%. Random Forest Classifier showed 65.28% accuracy. Gaussian NB had 62.5% accuracy. K Neighbors Classifier had least accuracy of 48.61% and it did not perform well.

Table 2 depicts the comparison of various algorithms defined by various parameters namely precision, recall and f1-score. The parameters are computed based on true and false positives, true and false negatives. Following are the cases to predict the result:

If the sample input data is negative then it is considered as True Negative (TN) provided if expected result is also negative.

If the sample input data is positive then it is considered as True Positive (TP) provided if expected result is also positive.

If the sample input data is positive then it is considered as False Negative (FN) provided if expected result is negative.

If the sample input data is negative then it is considered as False Positive (FP) provided if expected result is positive.

Multinomial Naïve Bayes The one of the algorithms from NLP which is based on probabilistic learning method is Multinomial Naive Bayes algorithm. The algorithm is influenced by the Bayes theorem and predicts the label of considered textual data. The probability of each label is calculated and the one which has the highest value is given as a result. This algorithm is a cluster of many algorithms in which all algorithms follow one standard rule that feature being classified is independent of other features because of which the presence/absence does not make any difference with presence/absence of other features.

The Table 3 shows the confusion matrix of Multinomial NB classifier. It showed True Positive as 15, True Negative as 39, False Positive as 13 and False Negative as 5. Hence Multinomial Naïve Bayes algorithm outperforms all other algorithms.

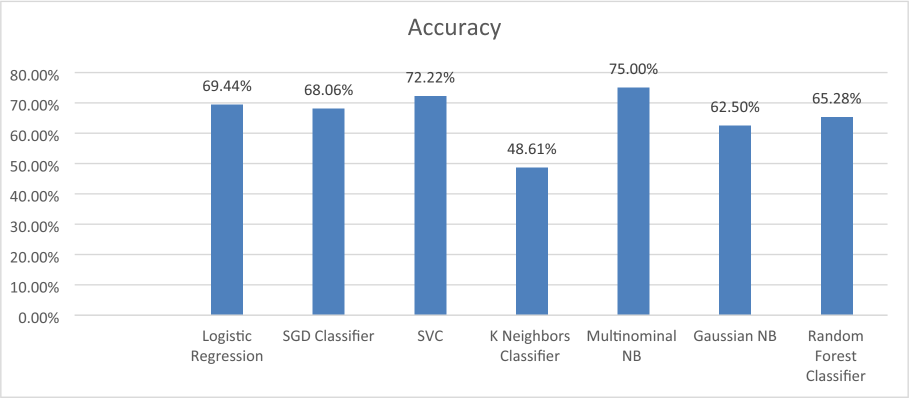


Fig. 14 Performance analysis of Machine Learning Algorithms on the collection of Kannada tweets. The algorithms that are considered here are Logistic Regression, SGD Classifier, SVC, K Neighbors Classifier, Multinominal NB, Gaussian NB, Random Forest Classifier

Table 2 Classification report of classifiers

SL.NO	Classifiers	Classification report				
1	Logistic Regression Classifier	Logistic Regression				
			Precision	Recall	F1-score	Support
		Neg	0.62	0.54	0.58	28
		Pos	0.73	0.80	0.76	44
		Accuracy	–	–	0.69	72
		Macro avg	0.68	0.67	0.67	72
		Weighted avg	0.69	0.69	0.69	72
2	Stochastic Gradient Descent Classifier	Stochastic Gradient Descent Classifier				
			Precision	Recall	F1-score	Support
		Neg	0.61	0.71	0.66	28
		Pos	0.79	0.70	0.75	44
		Accuracy	–	–	0.71	72
		Macro avg	0.70	0.71	0.70	72
		Weighted avg	0.72	0.71	0.71	72
3	Support Vector Classifier	Support Vector Classifier				
			Precision	Recall	F1-score	Support
		Neg	0.65	0.61	0.63	28
		Pos	0.76	0.80	0.78	44
		Accuracy	–	–	0.72	72
		Macro avg	0.71	0.72	0.70	72
		Weighted avg	0.72	0.72	0.72	72
4	K Neighbors Classifier	K Neighbors Classifier				
			Precision	Recall	F1-score	Support
		Neg	0.42	0.79	0.54	28
		Pos	0.68	0.30	0.41	44
		Accuracy	–	–	0.49	72
		Macro avg	0.55	0.54	0.48	72
		Weighted avg	0.58	0.49	0.46	72
5	Multinomial Naive Bayes Classifier	Multinomial Naive Bayes Classifier				
			Precision	Recall	F1-score	Support
		Neg	0.75	0.54	0.63	28
		Pos	0.75	0.89	0.81	44
		Accuracy	–	–	0.75	72
		Macro avg	0.75	0.71	0.72	72
		Weighted avg	0.75	0.75	0.74	72
6	Gaussian Naive Bayes Classifier	Gaussian Naive Bayes Classifier				
			Precision	Recall	F1-score	Support
		Neg	0.52	0.43	0.47	28
		Pos	0.67	0.75	0.71	44
		Accuracy	–	–	0.62	72
		Macro avg	0.60	0.59	0.59	72
		Weighted avg	0.61	0.62	0.62	72

Table 2 (continued)

SL.NO	Classifiers	Classification report				
7	Random Forest Classifier	Random Forest Classifier				
			Precision	Recall	F1-score	Support
		Neg	0.59	0.68	0.63	28
		Pos	0.78	0.70	0.74	44
		Accuracy	–	–	0.69	72
		Macro avg	0.68	0.69	0.69	72
		Weighted avg	0.70	0.69	0.70	72

3.2 Comparison with existing work

Table 4 shows the comparison of proposed model with existing models. From 14 existing models 6 models are purely processed in Kannada language only. One model in Tamil and Malayalam [1], another one in Hindi [2, 7] in Bengali, Hindi and Tamil, and in [10] Kannada is translated to English and then analysis is done. [4] consists the data in combination of both Kannada and English and sentiment analysis is done on both. [11] in English, Hindi and Bengali languages. [13] in English and Kannada cross languages and [12] in only English. [14] in only Telugu.

As compared to above models with only Kannada language, though [3] and [5] attained better accuracy, the data set with which they are experimented is very less. Our proposed model is experimented with 1000 Kannada tweets with 7 different algorithms such as Logistic Regression, Stochastic Gradient Descent Classifier, Support Vector Classifier, K Neighbors Classifiers, Multinominal Naïve Bayes, Gaussian Naïve Bayes and Random Forest Classifier and attained the accuracy of 75% with Multinomial Naive Bayes Classifier.

4 Conclusions

In this work, a classification using Machine learning is proposed to Sentiment Analysis of Twitter Feeds using Natural Language Processing in Kannada. In the modern days use of internet has been grown tremendously. Everything has moved online from shopping to education. Everyone is using social media for communication purpose. So, the data that is generated from social media every day is huge. Hence Sentiment Analysis plays an important role in determining the business insights and getting high finance payoff.

Table 3 Confusion matrix of multinominal naïve bayes algorithm

	Positive	Negative
Positive	15	13
Negative	5	39

Table 4 Comparison of proposed work with existing work

	Dataset used and size	Language considered for processing	Approach	Algorithm used	Accuracy
[1]	Tamil code mixed dataset with 11,335 comments and Malayalam code mixed data set with 4,851 comments	Code mixed data of Tamil and Malayalam	sub-word level model, a word embedding based model and Machine learning based architecture	Linear Support Vector and Logistic Regression	With Logistic Regression both Malayalam and Tamil attained F1 scores of 0.65
[2]	Data set size is not discussed	Hindi Language	Lexicon Based Approach (LBA) which is based on SentiWordNet Second approach is hybrid approach which is based on both unigram and Tf-Idf model	Naïve Bayes, Decision Tree, Support Vector Machine, Linear Regression, and Nearest Neighbor algorithms for tweets classification	92.97% with Decision Tree algorithm
[3]	Limited tweets as per author	Kannada, Hindi, Tamil, Telugu and Malayalam	TextBlob package from python in which the predefined categorized words are stored	Not Discussed in paper	98%
[4]	65,203 comments (combination of Both Kannada and English Comments)	Kannada-English code diverse dataset	To this code diverse dataset, the ML algorithms are applied	Decision Tree, K-Nearest Neighbors, Logistic Regression, Naïve Bayes, Random Forest and Support Vector Machine	Bi-LSTM depicts better ratings with average precision of 0.54, average recall of 0.51 and average F1-score of 0.54
[5]	500 positive and negative words each	Kannada	The model is experimented on 1000 words using ML algorithm	Decision Tree algorithm	85%
[6]	28 phone records with 32 features	Kannada	The model is experimented with 4 classes of words for sentiment analysis	Random Forest Ensemble algorithm	72%

Table 4 (continued)

	Dataset used and size	Language considered for processing	Approach	Algorithm used	Accuracy
[7]	999 tweets in Bengali language, 1222 tweets in Hindi language and 1103 tweets in Tamil language	Bengali, Hindi and Tamil	The model is experimented with the development data of 53 tweets in Bengali language, 56 tweets in Hindi language and no data is used for Tamil language	Multinominal Naïve Bayes, Logistic Regression, Decision Tree, Random Forest, Support Vector Machine (SVC) and Linear SVC	67% in Bengali language, 81.57% in Hindi language and 62.16% in Tamil language
[8]	100 movie reviews	Kannada language is translated to English	The algorithm is implemented on both Kannada and English translated text	Decision Tree algorithm	Kannada test data with precision of 0.78 and recall of 0.79. English test data with precision of 0.86 and recall 0f 0.67
[9]	182 positive and 105 negative reviews	Kannada language is translated to English	This model uses Turney's algorithm for translating Kannada reviews to English in dataset and then sentence level approaches are experimented on dataset	J48, Random Tree, ADT Tree, Breadth First, Naïve Bayes and support Vector Machine	The average precision under machine learning approach is 7.22%
[10]	Data set size is not discussed	Kannada language	sentiment analysis in Kannada language using Naïve Bayes Algorithm	Naïve Bayes Algorithm	65%
[11]	4,000 sample of reviews	English, Hindi and Bengali	Three algorithms are implemented on all three languages	simple neural network, convolutional neural network and long short-term memory neural network	84.1% with a model built with both long-short term neural network and convolutional neural network for all three languages together

Table 4 (continued)

	Dataset used and size	Language considered for processing	Approach	Algorithm used	Accuracy
[12]	1967 positive words and 783 negative words	English	This approach is experimented on English language for two candidates using different tools available for English language processing	AYLIEN package for named entity extraction and R, Rapid-Miner AYLIEN package for sentiment analysis	Candidate 1 is more popular than candidate 2
[13]	No Discussion of dataset size	English and Kananda	Model uses Bilingually constrained recursive auto encoder (BRAE) for sentiment analysis of two languages	supervised classifier	–
[14]	1400 Telugu sentences	Telugu	The proposed model is experimented in two phases, at first it focuses to classify the sentences as subjective or objective and later it classifies the subjective sentences as positive or negative	Sentiment classification using sentiWordNet	81% accuracy
Proposed model	1000 Tweets	Kannada language	comparative study of various machine learning algorithms for Kannada twitter sentimental analysis	Logistic Regression, Stochastic Gradient Descent Classifier, Support Vector Classifier, K Neighbors Classifiers, Multinominal Naïve Bayes, Gaussian Naïve Bayes and Random Forest Classifier	Multinomial Naive Bayes Classifier has performed better with accuracy of 75%

There are very large sentimental analysis models for English Language. But Sentiment Analysis in Kannada Language is very limited. Because of this we tried to give an efficient model for classifying twitter feeds using various classification techniques.

In our work, we have collected 1000 Kannada tweets and manually tagged them as Positive or Negative Sentence and used these for training our model. Before Classification, the preprocessing of data is very important. It reduces the dataset to greater level and increases model efficiency. Preprocessing include Tokenization, Data Cleaning, Removing Stop Words, Stemming. Once data is preprocessed the next step is feature extraction. We have used TF-IDF technique for feature extraction. In classification, we have used many classification algorithms such as Linear SVC, Logistic Regression, SGD, SVC, K-Neighbors, Multinomial NB, Gaussian Naïve Bayes and Random Forest Classifier.

Among all algorithms, Multinomial NB performed well and attained an accuracy of 75%. Gathering more data is still necessary because of model efficiency depends upon the data.

Acknowledgements Authors acknowledges Vision Group of Science and Technology (VGST), Govt. of Karnataka (Grant No. KSTePS/VGST-KFIST(L1)/2016-17 /GRD-570/2017-18/137/345/936)

Data availability Data will be available on reasonable request.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Mandalam AV, Sharma Y (2021) Sentiment analysis of Dravidian code mixed data. In: Proceedings of the first workshop on speech and language technologies for Dravidian languages. Association for Computational Linguistics, Kyiv, pp 46–54
2. Madan A, Ghose U (2021) Sentiment analysis for twitter data in the hindi language. 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence). pp 784–789. https://doi.org/10.1109/Confluence51648.2021.9377142
3. Rakshitha K, Ramalingam HM, Pavithra M, Advi HD, Hegde M (2021) Sentimental analysis of Indian regional languages on social media. Glob Trans Proc 2(2):414–420. https://doi.org/10.1016/j.gltp.2021.08.039. (ISSN 2666-285X)
4. Kannadaguli P (2021) A code-diverse kannada-english dataset For NLP based sentiment analysis applications. 2021 Sixth International Conference on Image Information Processing (ICIIP). pp 131-136. https://doi.org/10.1109/ICIIP53038.2021.9702548
5. Ranjitha P, Bhanu KN (2021) Improved sentiment analysis for dravidian language-kannada using dicision tree algorithm with efficient data dictionary. IOP Conference Series: Materials Science and Engineering, vol 1123. IOP Publishing
6. Hegde Y, Padma SK (2017) Sentiment analysis using random forest ensemble for mobile product reviews in Kannada. 2017 IEEE 7th International Advance Computing Conference (IACC). pp 777–782. https://doi.org/10.1109/IACC.2017.0160
7. Phani S, Lahiri S, Biswas A (2016) Sentiment analysis of tweets in three Indian languages. In: Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (WSSANLP2016). The COLING 2016 Organizing Committee, Osaka, Japan, pp 93–102
8. Rohini V, Thomas M, Latha CA (2016) Domain based sentiment analysis in regional Language-Kannada using machine learning algorithm. 2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT). pp 503–507. https://doi.org/10.1109/RTEICT.2016.7807872
9. Anil Kumar KM, Rajasimha N, Reddy M, Rajanarayana A, Nadgir K (2015) Analysis of users' sentiments from Kannada web documents. Procedia Comput Sci 54:247–256. https://doi.org/10.1016/j.procs.2015.06.029. (ISSN 1877-0509)

10. Hegde Y, Padma SK (2015) Sentiment analysis for Kannada using mobile product reviews: a case study. 2015 IEEE International Advance Computing Conference (IACC). pp 822-827. https://doi.org/10.1109/IADCC.2015.7154821
11. Bera A, Ghose MK, Pal DK (2021) Sentiment analysis of multilingual tweets based on Natural Language Processing (NLP). Int J Syst Dyn Appl (IJSDA) 10(4):1–12
12. Sharma A, Ghose U (2020) Sentimental analysis of twitter data with respect to general elections in India. Procedia Comput Sci 173:325–334
13. Impana P, Kallimani JS (2017) Cross-lingual sentiment analysis for Indian regional languages. 2017 International conference on electrical, electronics, communication, computer, and optimization techniques (ICEECCOT). IEEE
14. Naidu R et al (2017) Sentiment analysis using telugu sentiwordnet. 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET). IEEE
15. Fan G-F, Zhang L-Z, Yu M, Hong W-C, Dong S-Q (2022) Applications of random forest in multivariable response surface for short-term load forecasting. International J Electr Power Energy Syst 139:108073. https://doi.org/10.1016/j.ijepes.2022.108073. (ISSN 0142-0615)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.