Text Sentiment Analysis of Marathi Language in English And Devanagari Script

by

Harry Pradeep Gavali

Dissertation Submitted in partial fulfilment of the requirements for the degree

of

MSc in Data Analytics

At

Dublin Business School

Supervisor: Pierpaolo Dondio

January 2020

DECLARATION

I declare that this dissertation that I have submitted to Dublin Business School for the award of MSc in Data Analytics is my own research and original work; except where otherwise stated, where it is clearly acknowledged by references. Also, this work has not been submitted for any other degree and is entirely in compliance with Dublin Business School's academic policy.

Signed: Harry Pradeep Gavali

Dated: 6th January 2020.

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to my mentor and research supervisor Mr Pierpaolo Dondio, for guiding me on the right track from the beginning to the end of the research. I appreciate his constructive feedbacks and his willingness to always dedicate his precious time generously. He was always available to help, and his remarks have helped me a lot in order to make the research successful and professional.

I would also like to thank my parents, my friends and their parents who invested their precious time and helped me in compiling the data for the research. Few of them had to act as judges in order to rate the data, which was a crucial part of the research. Their help is deeply appreciated.

Furthermore, I would like to thank Google for allowing me to incorporate their Cloud Translation API which translated the data smoothly and swiftly.

Finally, I am indebted to my family for they have been a continuous support and have always motivated me throughout the completion of the thesis.

ABSTRACT

Marathi is a language spoken by a very large number of people in India and about 10% of the Indian population uses 'Marathi + English' while texting one another. This study focuses on text sentiment analysis of the mixed language text of Marathi (written in English) first, and then compares the accuracy again after the same sentences (written in Devanagari script) have been translated using Google's cloud translation services. Same machine learning techniques were applied on both in order to maintain equality. A new and accurate dataset which comprised of day to day sentences was compiled manually in order to reduce error. The outputs were later compared and the need to develop such researches further is highlighted. The results of the research show that the algorithms like Random Forest and SVM give us the highest accuracies of 65.41% and 64.16% respectively.

Keywords: machine learning, text classification, Marathi sentiment analysis, NLP Marathi.

Table of Contents

LIST OF FIGURES

Chapter 1: INTRODUCTION

Languages have existed for millions of years and since then have been going through a lot of advancements and changes. Over the course of history, we have seen language and culture being transferred and exchanged due to travellers visiting other shores and warriors capturing different lands. This led people of the land to learn the language of the other people. Those were the earliest instances of people learning different languages for convenience purposes. Fast-forward to today, with the invention of computers and the internet, the language and cultural barriers have been broken and it has become very easy to learn different culture and languages. With development of smartphones and equipping then with AI and machine learning, it has become possible to learn and even translate languages on the go. There is more data generated in the world today than it was since the invention of computers. And when we look at this data, it is in various formats and languages. Hence, the machines need to know what languages have been used and should be able to understand them. These problems fall under the category of Natural Language Processing. Today, Sentiment analysis has become one of the most researched fields. The research was fuelled due to the rise of social media and social media websites like WhatsApp, Twitter, Facebook, Instagram, etc. Sentiment Analysis is used o a very wide scale on such platforms in order to understand the consumer and sell them exactly what they desire. Sentiment analysis is also used to understand what people have to say on different matters that occur all around the world. Hence due to all these factors, the research on sentiment analysis has always been deeply done and is still continuing to improve.

As of today, we can see that Sentiment analysis of the English language has been done the most and has a lot of outcomes each with a unique efficiency and result. But as we know, cultural exchange has allowed people to be bilingual or even trilingual. In short, if people speak multiple languages, then it becomes natural that even the machines that are learning should learn all the languages. There has been a certain successful development even in other languages like German, Spanish, etc. and all with fairly successful results.

There are still certain areas which are not yet entirely covered or successfully researched in the field of sentiment analysis; and those fields are of the local languages; and of the languages that are spoken in one script but are written in some other script. India is a very big country which speaks a lot of languages. There are 22 official languages in India but more than a 1000 different mother tongues which result in more than 19500 dialects [1]. And creating a successful sentiment analysis model for all of them is nearly next to impossible – mostly due to the fact that not enough information is available on the unscheduled languages and that not a lot of people really use the language. Majority of the people in India use English words or sentences in their day to day lives. The languages have seemed to merge, and the combination is being used normally. Say for example Hindi; Hindi is the most common language used in India and spoken almost throughout the entire country. And now, Hinglish – a combination of Hindi and English, is the most widely spoken language in India. And obviously enough, Hinglish is also one of the most widely used form of textual language in India. Whereas when it comes to states, English, plus the states spoken language, seem to be the general mode of communication. Let us say for the state of Maharashtra, which has the native language of Marathi, the formula of 'English + Marathi' can be said to be the general mode of speaking as well as writing. Let us call it Mixed Language Text (MLT). It is in this field of local languages that we wish to develop our research further.

Marathi is a language spoken by the state of Maharashtra and hence is the local language of the state. But calling it local does not mean that a small population speaks it. Maharashtra is a state which is home to approximately 12.5 crore people of India [2]. That is approximately 10% people of India. And 10% is a huge number. Hence, that makes Marathi one of the most important and spoken language of the country. Hence, we have decided to use this as a local language for our study and research.

Sentiment Analysis for plain text written in only English, only Hindi or only Marathi is available with good and improving level of accuracy. However, as described earlier, plain text are not the language the masses use. Mixed scripts are the generally used formula, which means that the people do talk in Marathi but they type Marathi using the English (Latin) alphabet. Sentiment analysis of this code-mixed text is largely unexplored from a reseach point of view and it represents the focus of this research. The task is very challenges, since code-mixed text is where the grammar and language rules of two languages collide; making it difficult to create a stable and sensible sentiment analysis model with high precision and accuracy. Also, very little research has been done on such mixed text languages which makes it even more difficult for the research to be pursued.

In this research, we develop a sentiment analysis model for Marathi language written in English alphabet (MLT) and compared to a sentiment analysis model for the English language obtained by machine-translating Marathi language written in MLT into plain English. We want to check if a model directly based on Marathi language created in this research can outperform a sentiment analysis model based on the English language and operating on a machine translation of Marathi text. On the one hand, build a sentiment classifier in Marathi is challenging and there is very little prior work in the area, while on the other hand there are a large set of sentiment analysis tool for the English language with high accuracy. Our aim is to test to which degree the additional step of machine translation will decrease the accuracy of the English sentiment analysis model and how the resulting accuracy compare to the Marathi-only model.

The main objectives of the research can be said to be the following:

1. To analyse the sentiment of the Mixed Language Text (MLT) data using specific machine learning algorithms.
2. To translate the Marathi data to English text using the best available machine translation tools.
3. To analyse the sentiment of the obtained translated data and compare the results with those of the MLT results.

The scope of the research is to show what difficulties arise while trying to analyse the sentiment of MLT and how they can be addressed. Since there is no specific data available for this particular study, new datasets have been created with maximum accuracy and which can be used by the future generations in order to pursue the topic further.

During this research we created multiple datasets. One dataset contains all the Mixed Language Text sentences while the other dataset contains the same sentences written in Marathi. All the entries have been manually labelled with a sentiment score by multiple humans to guarantee high reliability of the resulting labels.

Our research questions are the following:

Research Question 1: which classification algorithm gives us the best result for the Mixed Language Text data?

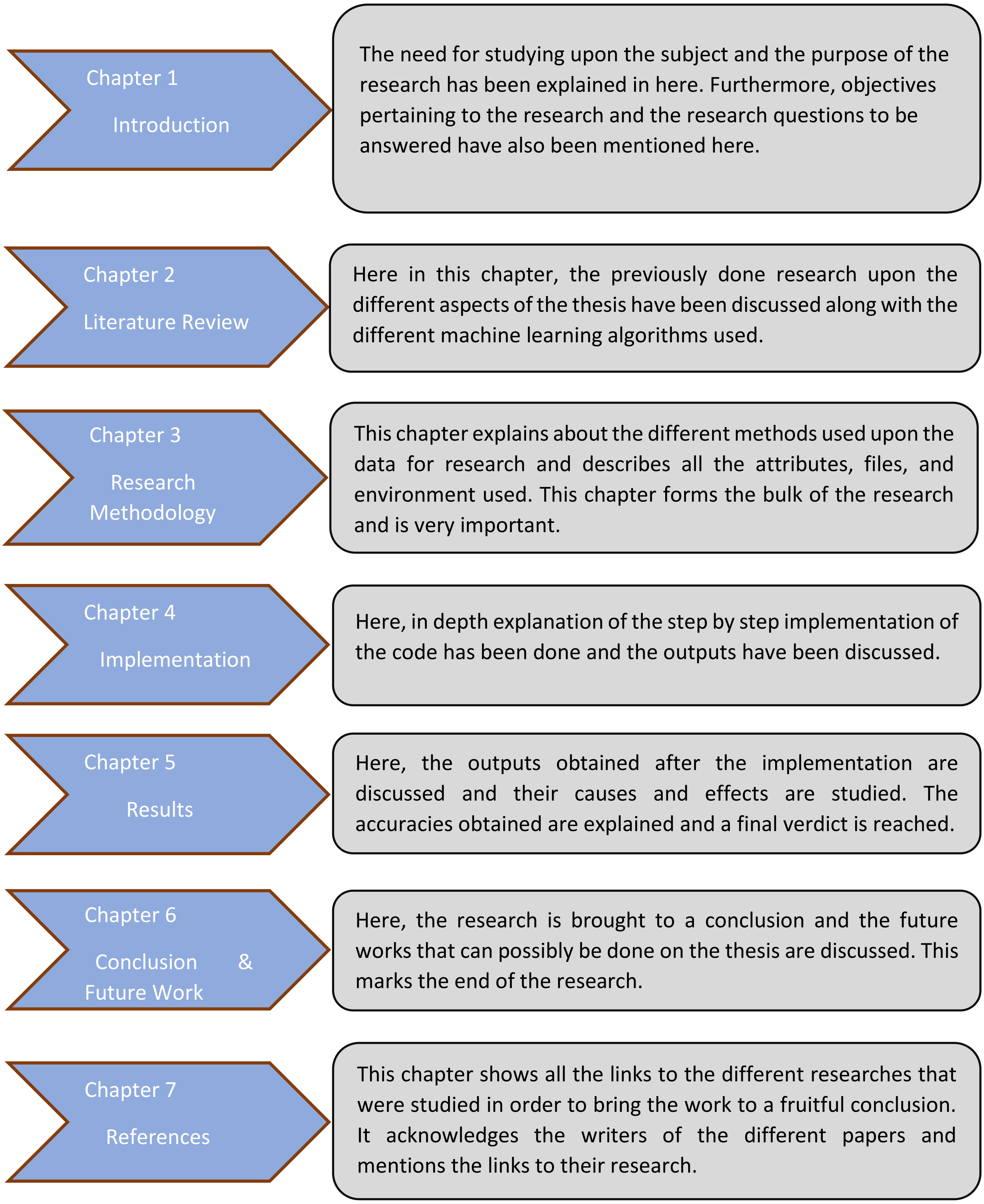
For the second research question, we rely on the translation provided by Google Translate[3], assuming that it can represent very well the state-of-the-art in machine translation and certainly the most used machine translation application.

Research Question 2: What is the accuracy of a sentiment analysis model for the English language operating on a machine translated dataset from Marathi language?

Research Question 3: Is the Marathi based model outperforms the model based on machine translation?

We have obtained the answers to our research questions and shall be discussed in the Conclusions section of the paper. Also, in order to give the research a very sensible flow, a dissertation roadmap has been used.

The dissertation roadmap is a way to tell the reader about the flow of the research and what he/she may find in that particular topic. Following is the dissertation roadmap for the thesis.



Chapter 1

Introduction

The need for studying upon the subject and the purpose of the research has been explained in here. Furthermore, objectives pertaining to the research and the research questions to be answered have also been mentioned here.

Chapter 2

Literature Review

Here in this chapter, the previously done research upon the different aspects of the thesis have been discussed along with the different machine learning algorithms used.

Chapter 3

Research Methodology

This chapter explains about the different methods used upon the data for research and describes all the attributes, files, and environment used. This chapter forms the bulk of the research and is very important.

Chapter 4

Implementation

Here, in depth explanation of the step by step implementation of the code has been done and the outputs have been discussed.

Chapter 5

Results

Here, the outputs obtained after the implementation are discussed and their causes and effects are studied. The accuracies obtained are explained and a final verdict is reached.

Chapter 6

Conclusion & Future Work

Here, the research is brought to a conclusion and the future works that can possibly be done on the thesis are discussed. This marks the end of the research.

Chapter 7

References

This chapter shows all the links to the different researches that were studied in order to bring the work to a fruitful conclusion. It acknowledges the writers of the different papers and mentions the links to their research.

Figure 1: Dissertation Roadmap

Chapter 2: LITERATURE REVIEW

Sentiment analysis has already been widely studied and is the centre of study for many researches. Every day there are organisations and people trying to better the understanding of sentiments which can be used by machines in order to learn and become more 'human'. In this section, the previously done work and accomplishments that led to this particular study have been discussed and reviewed.

E Mark Gold [4] Language identification practices and studies go way back into the 20th century where Gold tried to understand if just plain information(language structure) related to a text/language is enough for the user to understand which language is being used or that the text based rule detection is not always feasible and is dependent on the data and information along with the pre-defined rules of the given language to be identified.

Since then, a lot of work has gone into identifying languages and when the work was opened to Indian languages, various algorithms and studies were done in order to classify the text from regional languages, where **Kamal Nigam et al** [5] studied previously used algorithms for text classification and came to the conclusion that maximum entropy techniques are the way to go for text classification as the underlying principle for maximum entropy was that it preferred uniform distributions without any prior or external knowledge whatsoever. And since it was also a technique widely used in a variety of natural languages task, they ended up achieving up to 80% accuracy on some texts.

As the study into sentiment analysis from text classification developed more and more, opinion mining- a sub-discipline of text sentiment analysis came into picture. It was concerned with the feeling(opinion) a text document gave rather than the topic and subject of the document itself. Several researchers tried to determine whether a term that defines the opinion of the statement has a positive or negative connotation to it. Using this study, **Andrea Esuli et al.,** [6] created a SentiWordNet which was an adaptation to their previous synset classification method of determining the PN-polarity and SO-polarity of terms. They provided the world with a freely available WordNet which then was used by researchers for all their sense based lexical analyses.

Aditya Joshi et at., [7] In 2010, when the research of sentiment analysis was new to the Indian languages, Joshi, Balamurali and Bhattacharyya started their work in doing sentiment analysis for the language of Hindi. Their work was few of the earlier works done in Indian regional languages. They tried three different approaches and, in the process, developed a Hindi SentiWordNet for the language. They found that the analysis with HSWN performed better than machine translations and did worse than machine learning that came up in 2010

And while the researches were being carried out everywhere, **Heba Elfardy et al.,** [8] did their research based on determining whether applying sound-change rules affected the results of analysing the sentiment of the entire text. In the process, they coined the term Linguistic Code Switching (LCW) and helped future researchers in identifying how and where the code-switching may occur in dialect texts.

Akshat Bakliwal et al., [9] developed a big lexicon of adjectives and adverbs with polarity scores all while making use of the Hindi wordnet and also further developed the Hindi SWN by using linked wordnet analysis method. Their research and contributions have led to the conclusion that training machines with languages (machine learning) is not a better way than making them learn the sentiment (using LCW). Also, they developed a Hindi word corpus which consisted of product reviews which could be used for further research by others.

Bibekananda Kundu et al., [10] In their research paper authors have used statistical ways in order to detect foreign (Bengali) words in a mixed coded script. They stated that using statistical methods was language independent and that their code could be used for other languages also. This was a new take on sentiment analysis as until now only code-mixing and code switching was used which caused errors like machine translation errors. They achieved an accuracy of 71.82 percent on their used Bengali-English mixed code texts.

Yogarish Vyas et al., [11] did their work on the tougher ares of language understanding, which is POS Tagging. In their paper, they have supported the fact that POS tagging added to the benefits of sentiment analysis of transliterated texts and that language identification and transliteration are the two major challenges that impact POS tagging accuracy. Although their work provides great insight and definitely helps in understanding the Sentiment analysis process and difficulties arising in it clearly, it has been observed that not using POS tagging also gave similar results in newer researches and that using the tiring method of POS tagging was not a necessity.

Ben King et al., [12] used weakly supervised methods on their data which they said 'could be called minority language data as any language can be considered monority if the data is small', and found that CRF trained with GE was the most accurate method and gave an accuracy of about 90 percent. And that accuracy was for any language since they did not use any conventional methods like lexicon usage and POS tagging. Also, they concluded that handling named entities was a major challenge for the system and that creating a separate label for named entities would be a good option for the same.

Utsab Barman et al., [13] did research on automatic language identification with Indian language code mixing from social media communication and concluded that the method of supervised learning surpasses that of the dictionary-based learning method. But the dictionary could be used to add features to the supervised learning method and make It more efficient. This all was achieved and yet they have not made the use of word-level code-mixing and has been left for future works.

It can be observed that for decades, work has been done in analysing the language of English and it has been only a few years that text sentiment analysis was researched in the local languages of other countries. Considerable work has also been done in the language of Hindi with a good and capable Hindi SentiWordNet (HSWN) available for study and research. But very little work has been done in the field of Marathi language, and **Priyanka Pradip Kulkarni et al.,** [14] The authors of this study took it a step ahead by identifying written word identification for Marathi and Sanskrit languages using the genetic algorithm. They trained and tested a lot of inputs and achieved a good accuracy which could be made better by training more examples.

K Ravi et al., [15] did their study of sentiment analysis on Hinglish text using different combinations of feature selection methods and found that a combination of TF-IDF, GR and RBFNN was the best way to classify Hinglish texts. They also suggested that a lexicon-based approach may be tried in order to generate more sensitivity.

Muhammad Bilal et al., [16] proposed three different classification models for text classification and Sentiment analysis using the WEKA which is the Waikato Environment for Knowledge Analysis. The idea behind this paper was to analyse the sentiments from extracted blog which are opinions written in Roman-Urdu language and also English. Various machine learning algorithms like Naïve Bayesian, Decision Tree and KNN were applied thus, the best model according to this research paper was the KNN as the accuracy was more than other models as well as the precision and F-measure was more when compared with other models.

H kaur et al., [17] did a dictionary-based sentiment analysis of hinglish text for they stated that translations for hinglish texts are never 100% accurate and hamper accuracy. They came up with two dictionaries, one for English and one for Hindi data which were capable of handling word variations and case sensitivity.

Tripto et al., [18] proposed complete understanding about the various techniques used to classify sentiments. In this particular study three different languages text were used for analysing the sentiment, which is Bangla, English and Romanized. They used deep learning model for sentiment analysis of Bangla sentences and categorized data into three-class label which were positive, neutral and negative. Furthermore, they have also labelled the data into five class which were strongly positive, positive, neutral, negative, strongly negative. Thus, according to this study, the result obtained for both the classified labelled class was more than other approaches with the accuracy of 65.97% and 54.24% respectively

Marathi language is spoken by a large part of India and hence application and study of sentiment analysis in the language was expected. Marathi is written in the Devanagri script, and **Sujata Deshmukh et al.,** [19] did their research on this script of Marathi language. So, their research deals with the original devanagri script written Marathi. They supported the use of a corpus-based approach complete with word polarity and gave the output in the format of a cumulative polarity score of the input sentence. They achieved good results limited to their available corpus – which was also their stated issue that the resources to study the language were limited.

The part that was still very fresh and barely untouched was determining the sentiment analysis of transliterated and code-mixed Marathi language. Marathi written in Roman script has become immensely popular and widely used. Little to no research was found for the same. **Mohammed Arshad et al.,** [20] A brilliant effort was taken by both of the authors where they analysed the sentiment of such mixed code transliterated Hindi and Marathi texts. Since data was quite abundantly available now due to social media and communication, they worked on Hindi language more than Marathi and used supervised learning methods to classify the languages. They also led to a start to create a Marathi SentiWordNet but it is quite basic and in its developing stage. They concluded that the context affected the statement sentiment analysis to a large extent and that it should be taken into consideration in future works.

Abhishek Kaushik et al., [21] Machine learning algorithms for text sentiment analysis follow a proper structure. Firstly, dataset is divided into a training and testing set. The training dataset is generally considered to be 70% of the entire dataset. It has the featured vectors and their related labels. Next, a classification model is developed. This classification model then works on the train set and then classifies the vectors and their related labels. After this, the resulting model is put into validation on the test set (remaining 30% of the dataset) for which the classifier had not seen the related labels. The accuracy with which the classifier validates the test set shows how good the model was built. The machine learning algorithms used in this particular research are Random Forest Algorithm, Decision Tree Algorithm, Support Vector Machine Algorithm, Logistic Regression Algorithm, KNN Algorithm and Naïve Bayes algorithm.

Chapter 3: METHODOLOGY

Every research needs to follow a certain methodology to reach their conclusions. Here in this section, use of a slight modification of the CRISP-DM process has been made, for the addition of the data gathering step and deletion of the business understanding step was required for this particular research. Hence, the following methodological steps have been followed:

1. Obtaining Data: To make the research strong and avoid the involvement of crude language and spellings from different dialects of the same language, the database has been created manually with the highest levels of accuracy.
2. Sentiment valuation: a few participants were asked to rate the sentences on a simple scale in order to give an unbiased sentiment value to the sentences.
3. Data Preprocessing: Once the data was available, a few preprocessing steps were applied which helped the data to become cleaner and more suitable to use for the research. The steps applied were stopword elimination, null value elimination, digit, special characters and punctuation elimination, formatting the data to convert all the alphabets to the same case (lowercase in this scenario), and finally tokenization of the sentences.
4. Machine learning: The data was divided into a 70:30 ratio of which 30% was used for testing and 70% was used for training the models. Various classification machine learning algorithms were applied, and their outputs were obtained.
5. Result: accuracies were obtained after successfully running the machine learning algorithms.
6. Statistical Testing: the different algorithms were tested against one other to obtain their confidence levels and the outputs are displayed and explained.

3.1 Datasets:

A proper dataset was not available for the research at hand, hence, a manually written dataset was chosen to be the best choice as it allowed for the work to be done with much better precision. A participant-based data collection was tried and put to use, but after discovering that multiple participants had different dialects which resulted in different spellings for the same words, a different method of data acquiring was given thought. Data was manually typed while keeping in mind that a similar number of positive, negative and neutral sentiment sentences were required for the study. Two datasets were created, one of which had the mixed language text typed sentences while the other had the same sentences written in Marathi. This also allows to be confirmed that the datasets are highly accurate and have negligible errors.

The Marathi data has also been typed by the research author (Marathi being his mother tongue) and that has enabled for the data to be highly accurate with the best examples that can be used for the research. This allowed the data to have two columns- one for the sentences (labelled as 'comment'), and one for the sentiment values (labelled as rating).

The dataset containing the mixed language text sentences has been saved as a file with a .csv extension with name as 'tdata' whereas the dataset containing the same sentences in Marathi has been saved as a file named 'marathi data' with the same extension.

This dataset can also be said to be a contribution to the data science community in a way that it will be provided openly (open source) and can be used by anyone who wishes to pursue the topic further or improve upon the same research.

3.2 Sentiment Valuation:

After the dataset was built, it was supplied to 8 educated native Marathi speakers (including the research author), who then rated the sentences on a 3-class scale. They were asked to rate all the positive sentences with a +1 (positive one), negative sentences with a -1 (negative one), and the neutral sentences with a 0 (zero). On inspecting the obtained answers, it was evident that maximum of the sentences was rated the same by all eight participants, with some sentences having a different answer from some of the participants.

The answers were compared, and it was decided that to label one unique value to the particular sentence, the method of mode was to be followed. This gave the sentence a value that was selected by a majority of the participants, thereby making sure that there was no bias, and concluding that the sentiment value given by majority of the participants is the right sentiment value.

3.3 Data Preprocessing:

The data obtained needs to be made uniform and clean in order to enable smooth and noise free implementation. For the same purpose, few data preprocessing steps were applied. Firstly, the stop-words were removed. Stop-words are the words which occur most commonly in a document and give no benefit or advantage in determining the sentiment of the document. They just increase the word count and may increase the workload on the program or software. They can be said to be the opposite of keywords; where keywords are the most important words in any document, stop-words are the most unimportant or non-utilizable words in the document. Examples of stop-words are A, an, the, he, she, so, that, can, etc. Examples of stop-words in Marathi are तो (he), तू (you), आणि (and), ती (she), etc.

A separate file containing the stop words for mixed language text Marathi was also created manually in order to keep accuracies to the higher end. This file was then pasted in the nltk corpus stop words directory. The same was done English stop words as it was noticed that there were some words in the English stop words file which affected the efficiency of the translated texts.

After the stop words were removed, the removal of null characters and invisible characters was required. Invisible characters can be spacebars or tabs. They increase error margin while running the code and hence had to be removed. Now, it is also known that digits and number values play no part in analyzing the document sentiment, hence even they had to be eliminated. And after the data has been cleansed of them, special characters and logos remain. Special characters are commas, full stops, exclamation marks, etc. Although special characters help the reader to understand the texts better, they play no part for the machine; keeping them makes no sense to the code and hence they had to be removed.

Once the data is clean of all the extra materials, it is important for the text to be in the same format and style, for same words in different written styles (capitalization at the first word or random word being capitalized) mean different words to the machine. Hence the conversion of every word to lowercase was done. This helped in streamlining the document and be more accurate.

Once data cleaning was complete, the data was tokenized. Tokenization can be said to be the process wherein the document or sentences are broken up or split into smaller meaningful elements like words, keywords, phrases, etc. While tokens can be anything ranging from symbols to complete sentences, the tokens generated for the research dataset are in the form of single words separated by spaces. Tokenization is an important step for the research as these forms a base for performing further sentiment analysis by making it easier for the machine to learn the representation each word brings while determining the sentiment of the sentence.

3.4 Machine Learning:

Machine learning has certain points which are made a note of before beginning any algorithm or applying any model. Firstly, it is checked if the data is supervised or unsupervised. Supervised data basically means the data that has labels and annotations which can help us understand and learn about the data; unsupervised data refers to the data that does not any have labels or annotations and gives us no idea about what the data is about. For our research, the data is supervised data and that concludes that supervised machine learning algorithms are to be used. Next comes the part where it is checked if Regression or Classification algorithms are to be used. Regression algorithms are used where the prediction of continuous values is done or where the correlation between two or more independent variables is to be done. Examples of regression can be time series predictions, weather predictions, etc. Classification algorithms are used where prediction or classification of discrete and non-continuous values is to be done into groups or other forms using certain parameters. Examples of classification can be speech and text classification, email spam classification, gender classification etc. Since the research data demands predictions in a 3-class system, we can conclude that use of classification algorithms is to be made.

Before beginning with the machine learning algorithms, it is important to know and understand about the environments and software and tools that will be used for the research. The coding language of python has been used here. Python provides a large resource of libraries when it comes to text sentiment analysis. Also, python is platform independent and has a very big community which makes it the most widely used programming language in the world today. The ease of use of python is also an added advantage. The use of Jupyter Notebook has been done for typing and executing the python code for it allows live code to be shared and edited online and shows the outputs code block by code block and is easy to use and compile. The jupyter notebook has been used from the anaconda platform which is a widely used data science distribution which leads in open source innovation for machine learning. Anaconda comes with a lot of pre-installed requisites that are helpful for languages like R and python.

When machine learning is being done, it is a common process to split the data into train sets and tests sets. Train set is the data or part of the original data that is used for training the chosen machine learning algorithms. Test set is the data or part of data that is used for assessing or checking the performance of the chosen machine learning algorithms after they have shown promising results while performing on the train set. Usually, a ratio of 70:30 (70% data for training and 30% data for testing) is chosen because training the algorithm with more data tends to make the chosen model better at doing the task at hand. Once the data is split correctly, the application of different machine learning algorithms is done.

This can be said to be the core of the research as it is here that different algorithms are being applied on the data which are then compared, and their outputs are interpreted for the research understanding. The following algorithms are being applied for the research data.

1. Random Forest Algorithm
2. K-Nearest Neighbor Algorithm
3. Naïve Bayes Algorithm

4. Decision Tree Algorithm
5. Support Vector Machine Algorithm
6. Logistic Regression Algorithm

The above-mentioned algorithms have a very high preference among data scientists and show a very high level of performance and have been chosen for the research study.

1. Random Forest Algorithm: This algorithm is an advanced version of decision tree algorithm which is explained later below. Simply put, Random Forest is made up of lots of decision trees. And while it samples all the trees, it at the same time applies random training points to each of the tree and also considers random subsets of features while splitting the nodes. This gives hundreds and thousands of different outcomes and due to this, it gives a higher accuracy.

2. K-Nearest Neighbor Algorithm: This algorithm is one of the simplest yet highly effective algorithm. Simply put, what KNN does is that it takes the newly given data or point and then matches its similarity with the available cases (data it was trained on) and then based on similarity decides which category to put the new case data in. When the training data is given to KNN, it stores the data instead of learning on it and then just uses it to match similarities with the new data. Hence it is also called as the lazy learner algorithm.

3. Naïve Bayes Algorithm: The most common and simple and highly effective algorithm when it comes to machine learning especially text classification, Naïve Bayes is very simple to understand. It is based on the probabilities of the event and uses Bayes Theorem to implement it. It uses posterior, likelihood, prior and marginal probabilities to find out the predictions of the dataset and works best on highly dimensional datasets. Spam filtering and Sentiment analysis are some of the most common fields where Naïve Bayes is used. It looks at every individual attribute/dimension individually in order to predict the outcome. For

example, for a vehicle, if it has four wheels, has a roof, has a low ground clearance, then naïve bayes will classify it to be a sedan type of car. Whereas if all the features of previous case are kept same and instead of low ground clearance, the ground clearance is high, then it will be classified as an SUV.

4. Decision Tree Algorithm: As the name suggests, it is a tree-structured classifier, and contains nodes and branches in order to reach a decision. It has decision nodes and leaf nodes which are used to make decision rules and display outcomes for the same respectively and has branches which are nothing, but the decision rules applied in order to reach the decision or leaf node. The CART algorithm is used to construct a decision tree (CART: Classification and Regression tree Algorithm) and is one of the easiest algorithms to understand for it depends almost on the idea of how humans think while making decisions.

5. Support Vector Machine Algorithm: SVM algorithm works on the principle of margins and hyperplanes. Unlike KNN, SVM creates decision boundaries by using the farthest apart points of distinctive datasets as margins and then drawing a line which distinctively divides the given classes/datasets and helps them classify. This line is called the hyperplane. There can be many hyperplanes for a given problem, SVMs task is to find the best hyperplane to reduce maximum errors while classifying new dataset. The points that lay on the margin are called support vectors, and because they help in drawing an optimal line between classes, the name Support Vector Machine was given.

6. Logistic Regression Algorithm: Although it contains regression in its name, Logistic Regression is a classification algorithm. Its name is based upon linear regression for both have the same underlying principle. And whereas linear regression has continuous output points, logistic regression gives us a probability value after putting the obtained output through a

sigmoid function. It is a very special kind of algorithm because it uses continuous and discrete datasets, applies regression techniques on it, applies a sigmoid function and gives a curved line which predicts probabilistic values between 0 and 1. Thus, because it classifies objects and data even though it applies predictive regression, logistic regression is a very powerful algorithm used in classification.

Result

The results obtained from the machine learning algorithm are then put into a table and their values are compared. Also, the confusion matrix of the algorithms are compared which helps in explaining how and why the outputs deferred.

3.5 Statistical Testing:

In order to compare models between them, we performed a statistical test on pair of models. This is to check if the difference between two models can be due to chance or it is highly likely that the models are actually different in terms of accuracy. To remove this conclusion, statistical testing is done at different confidence levels. Statistical testing is a way to remove any doubt that the outputs obtained were just by luck and also shows the confidence with which it can be said. The statistical test used was the T-test, and the formula is as follows:

$$t = (acc_1 - acc_2) / \sqrt{\frac{acc_1(100 - acc_1)}{n} + \frac{acc_2(100 - acc_2)}{n}}$$

where acc_1 and acc_2 are the accuracies of the two models to be compared and n is the total number of data values. The models are statistically different if the value of t is higher than t_x, the critical value for the selected confidence level x.

The below diagram shows the general flow of the research and how the two different datasets are compared. It can be said to be the workflow diagram for the research.

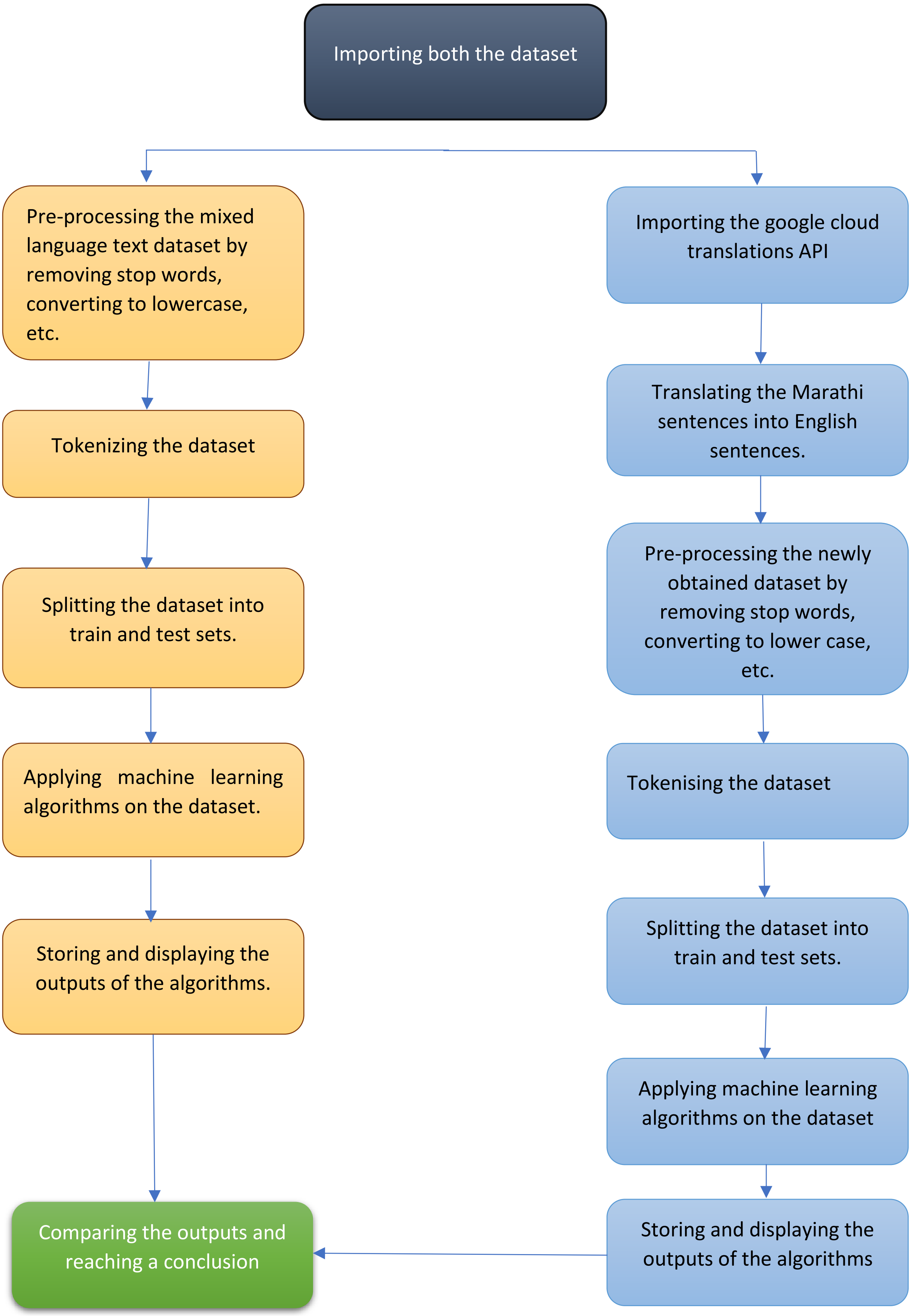

Importing both the dataset
Pre-processing the mixed language text dataset by removing stop words, converting to lowercase, etc.
Tokenizing the dataset
Splitting the dataset into train and test sets.
Applying machine learning algorithms on the dataset.
Storing and displaying the outputs of the algorithms.
Comparing the outputs and reaching a conclusion
Importing the google cloud translations API
Translating the Marathi sentences into English sentences.
Pre-processing the newly obtained dataset by removing stop words, converting to lower case, etc.
Tokenising the dataset
Splitting the dataset into train and test sets.
Applying machine learning algorithms on the dataset
Storing and displaying the outputs of the algorithms

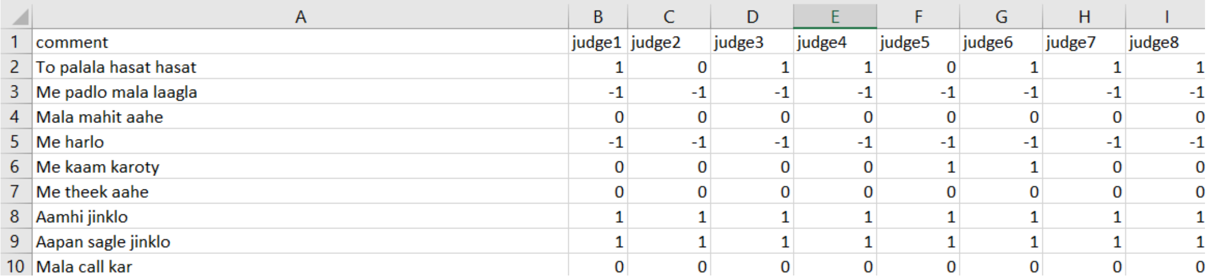
Figure 2: Conceptual diagram of the research process

Chapter 4: IMPLEMENTATION

In this section, step by step explanation has been given which has helped take the research to its fruitful conclusion. The steps mentioned in the methodology have been applied to the research dataset and their outputs have been displayed. The outputs of the machine learning algorithm have been discussed in the results section of the paper.

4.1 Importing and understanding the dataset:

The initial dataset contains 9 columns which are as follows:

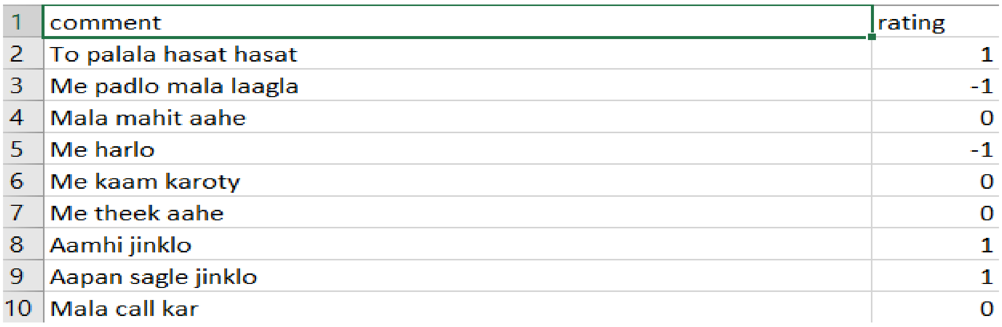


	A	B	C	D	E	F	G	H	I
1	comment	judge1	judge2	judge3	judge4	judge5	judge6	judge7	judge8
2	To palala hasat hasat	1	0	1	1	0	1	1	1
3	Me padlo mala laagla	-1	-1	-1	-1	-1	-1	-1	-1
4	Mala mahit aahe	0	0	0	0	0	0	0	0
5	Me harlo	-1	-1	-1	-1	-1	-1	-1	-1
6	Me kaam karoty	0	0	0	0	1	1	0	0
7	Me theek aahe	0	0	0	0	0	0	0	0
8	Aamhi jinklo	1	1	1	1	1	1	1	1
9	Aapan sagle jinklo	1	1	1	1	1	1	1	1
10	Mala call kar	0	0	0	0	0	0	0	0

Figure 3: Initial dataset

‘comments’ are the mixed language texts which are the basis of the research whereas judge1 – judge8 are the independent participants who rated the sentences based on their perception and understanding of the sentence.

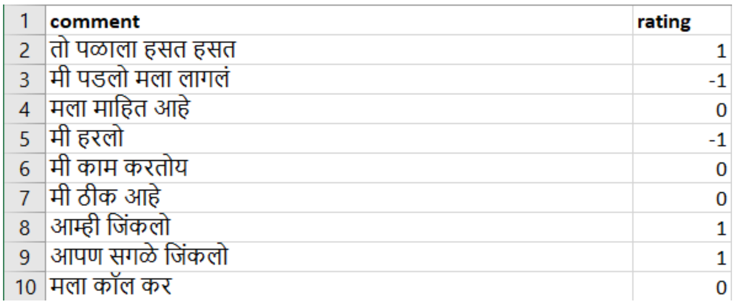
Later on, the mode (maximum occurring value) of rating for each of the sentiment was chosen to be the final rating and that gave the first dataset file for the research. The file was named ‘tdata.csv’



1	comment	rating
2	To palala hasat hasat	1
3	Me padlo mala laagla	-1
4	Mala mahit aahe	0
5	Me harlo	-1
6	Me kaam karoty	0
7	Me theek aahe	0
8	Aamhi jinklo	1
9	Aapan sagle jinklo	1
10	Mala call kar	0

Figure 4: Final Rating

The second dataset file is named 'marathi data.csv' and has the same sentences as the first file, but they are written in Devanagari or Marathi script. The ratings are same for them because the sentences are the same.



1	**comment**	**rating**
2	तो पळाला हसत हसत	1
3	मी पडलो मला लागलं	-1
4	मला माहित आहे	0
5	मी हरलो	-1
6	मी काम करतोय	0
7	मी ठीक आहे	0
8	आम्ही जिंकलो	1
9	आपण सगळे जिंकलो	1
10	मला कॉल कर	0

Figure 5: Final rating for Marathi dataset

4.2 Pre-processing the data:

After the files are imported, data pre-processing was done. The following graph shows the different classes and the number of entries per class for the dataset. It can be observed that rating 0 (neutral sentiment) sentences are the most. But on an overall scale, it can be said that the quantities of sentences for the different ratings are almost the same.

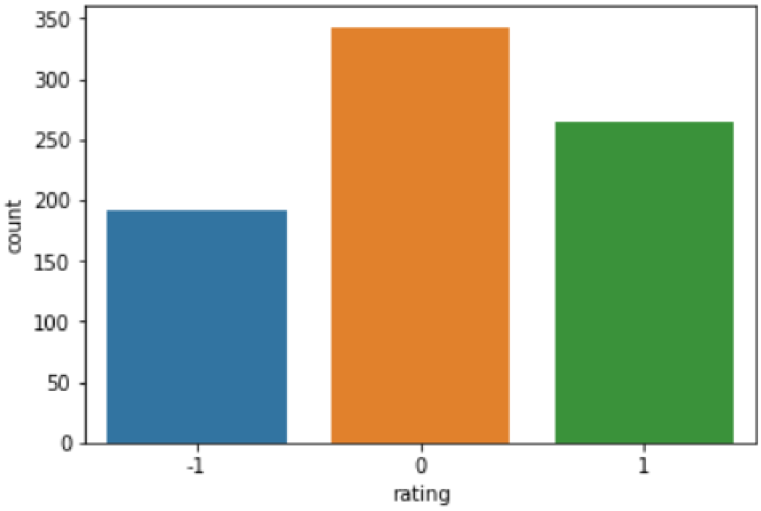
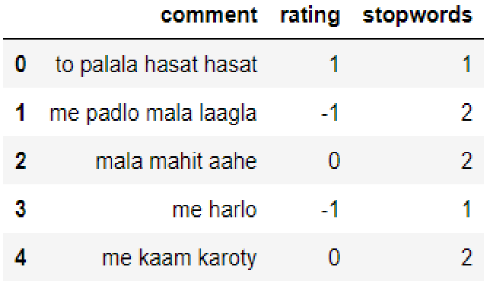


Figure 6: Graph showing different classes and their count

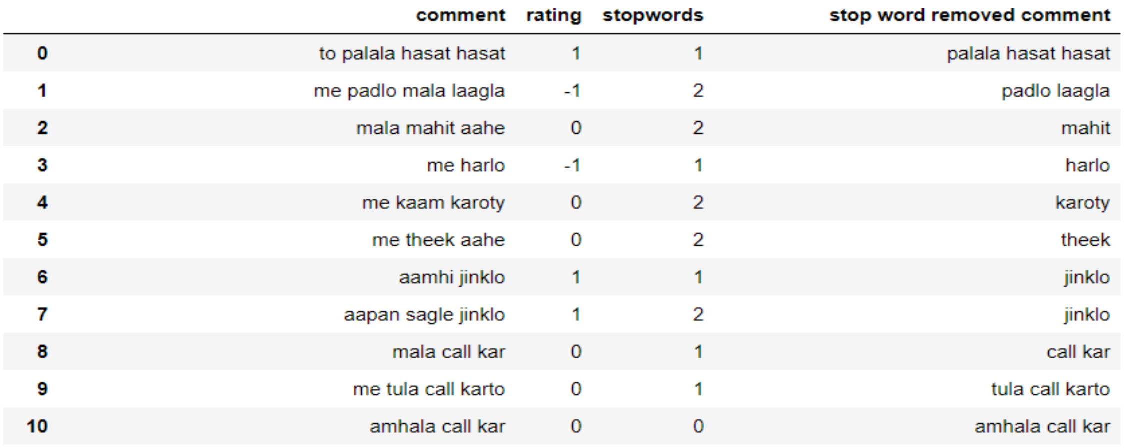
Now, it was required to turn the data into lowercase to facilitate smooth execution of the machine learning algorithms. That was done and then the number of Marathi stop words in each sentence was recorded. The stop word file for the mixed language text was manually created which allowed foe maximum accuracy. The following image shows the two methods done.



	comment	rating	stopwords
0	to palala hasat hasat	1	1
1	me padlo mala laagla	-1	2
2	mala mahit aahe	0	2
3	me harlo	-1	1
4	me kaam karoty	0	2

Figure 7: Number of stop words in every sentence

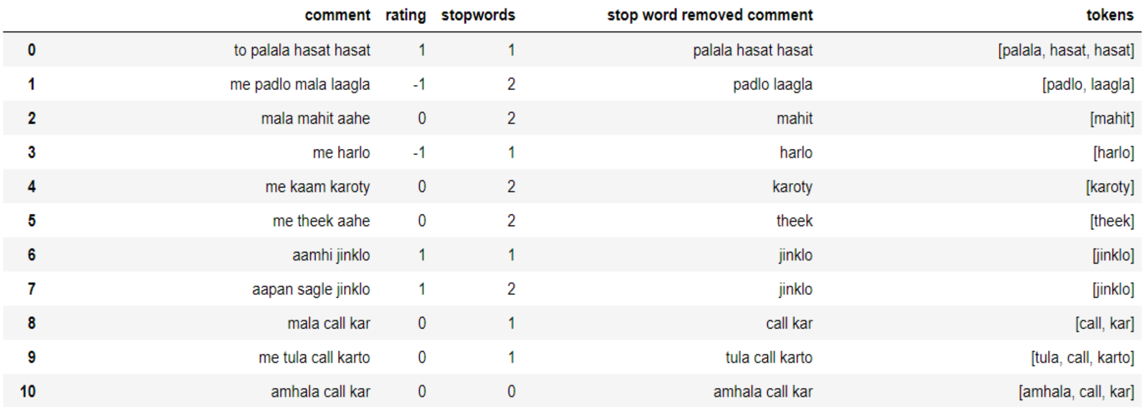
Next, after viewing the stop words, they had to be eliminated and the sentences were displayed again. It can be seen how only the important words remain in the output. This forms the base on which the algorithms will be implemented.



	comment	rating	stopwords	stop word removed comment
0	to palala hasat hasat	1	1	palala hasat hasat
1	me padlo mala laagla	-1	2	padlo laagla
2	mala mahit aahe	0	2	mahit
3	me harlo	-1	1	harlo
4	me kaam karoty	0	2	karoty
5	me theek aahe	0	2	theek
6	aamhi jinklo	1	1	jinklo
7	aapan sagle jinklo	1	2	jinklo
8	mala call kar	0	1	call kar
9	me tula call karto	0	1	tula call karto
10	amhala call kar	0	0	amhala call kar

Figure 8: Sentences with stop words removed

On obtaining the stop word removed sentences, the next step is to convert them into tokens. By that, it means that each sentence will be converted into a list of tokens – in this case single words. This allows for smoother and better outputs from the algorithms. The following image shows the same.



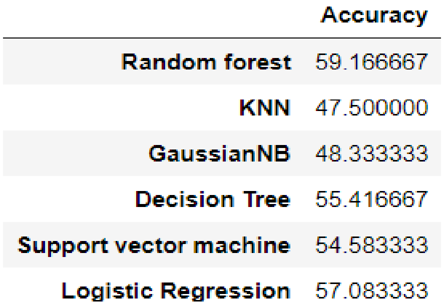
	comment	rating	stopwords	stop word removed comment	tokens
0	to palala hasat hasat	1	1	palala hasat hasat	[palala, hasat, hasat]
1	me padlo mala laagla	-1	2	padlo laagla	[padlo, laagla]
2	mala mahit aahe	0	2	mahit	[mahit]
3	me harlo	-1	1	harlo	[harlo]
4	me kaam karoty	0	2	karoty	[karoty]
5	me theek aahe	0	2	theek	[theek]
6	aamhi jinklo	1	1	jinklo	[jinklo]
7	aapan sagle jinklo	1	2	jinklo	[jinklo]
8	mala call kar	0	1	call kar	[call, kar]
9	me tula call karto	0	1	tula call karto	[tula, call, karto]
10	amhala call kar	0	0	amhala call kar	[amhala, call, kar]

Figure 9: Tokenized sentences

4.3 Applying machine learning:

After this step, data was split into the ratio of 70:30 train-test sets which were later utilised to run the machine learning algorithms on. The algorithms are trained on the 70% train data and then the test set data is used to see how the algorithms fare.

The following were the outputs of the machine learning algorithms:

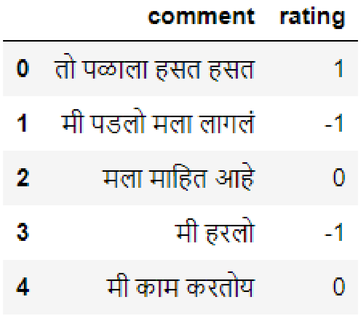


	Accuracy
Random forest	59.166667
KNN	47.500000
GaussianNB	48.333333
Decision Tree	55.416667
Support vector machine	54.583333
Logistic Regression	57.083333

Figure 10: o/p table for the algorithms

This is output for the dataset of the mixed script language. Next, we proceed to the Marathi text dataset and carry forward the research.

The Marathi dataset was imported and then displayed as follows:

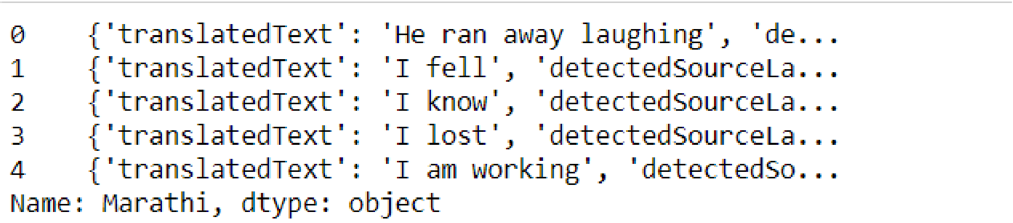


	comment	rating
0	तो पळाला हसत हसत	1
1	मी पडलो मला लागलं	-1
2	मला माहित आहे	0
3	मी हरलो	-1
4	मी काम करतोय	0

Figure 11: Final Marathi data

Now comes the important part - google cloud services were imported for the purpose of translating the texts from Marathi to English. For the same, google cloud account was created to avail their services. The setup was completed which provided a json API key for using the cloud translations.

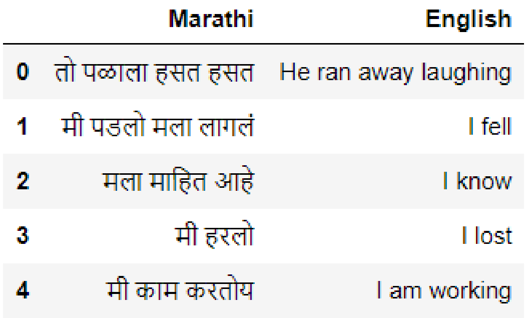
After translations, the following output was obtained which was then saved as a new file named ‘mdata.csv’



```
0    {'translatedText': 'He ran away laughing', 'de...
1    {'translatedText': 'I fell', 'detectedSourceLa...
2    {'translatedText': 'I know', 'detectedSourceLa...
3    {'translatedText': 'I lost', 'detectedSourceLa...
4    {'translatedText': 'I am working', 'detectedSo...
Name: Marathi, dtype: object
```

Figure 12: Translated text obtained after using google services

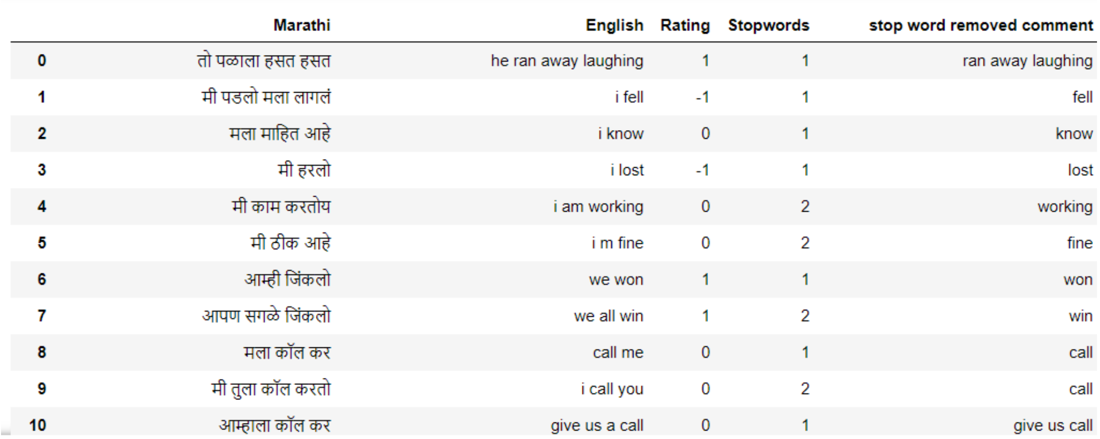
The file was then displayed after checking if there was any need for manually checking the file for errors. The file obtained as end result was as follows:



	Marathi	English
0	तो पळाला हसत हसत	He ran away laughing
1	मी पडलो मला लागलं	I fell
2	मला माहित आहे	I know
3	मी हरलो	I lost
4	मी काम करतोय	I am working

Figure 13: Final translated text from Marathi to English

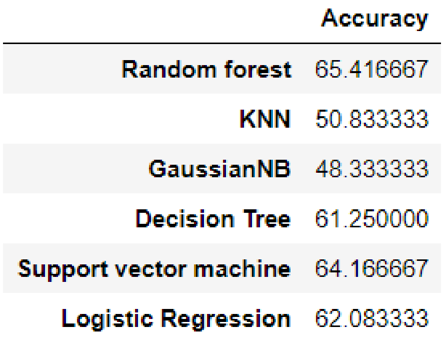
As can be seen, the dataset again needs to be cleaned in the manner that was done for the mixed language text data. After cleaning the data, removing the English stop words and special characters and after turning the data to lowercase, the following output that also shows the stop word removed sentences was obtained:



	Marathi	English	Rating	Stopwords	stop word removed comment
0	तो पळाला हसत हसत	he ran away laughing	1	1	ran away laughing
1	मी पडलो मला लागलं	i fell	-1	1	fell
2	मला माहित आहे	i know	0	1	know
3	मी हरलो	i lost	-1	1	lost
4	मी काम करतोय	i am working	0	2	working
5	मी ठीक आहे	i m fine	0	2	fine
6	आम्ही जिंकलो	we won	1	1	won
7	आपण सगळे जिंकलो	we all win	1	2	win
8	मला कॉल कर	call me	0	1	call
9	मी तुला कॉल करतो	i call you	0	2	call
10	आम्हाला कॉल कर	give us a call	0	1	give us call

Figure 14: Marathi data after translation and removing stop words

After this, the sentences were tokenized, the data was then split into the train and test sets and machine learning algorithms were applied on them. It can be seen that certain sentences were left with only one word after removing stop words. This will definitely make a different impact on the outputs obtained. The following table shows the output obtained for the translated text dataset.



	Accuracy
Random forest	65.416667
KNN	50.833333
GaussianNB	48.333333
Decision Tree	61.250000
Support vector machine	64.166667
Logistic Regression	62.083333

Figure 15: o/p table for translated Marathi dataset

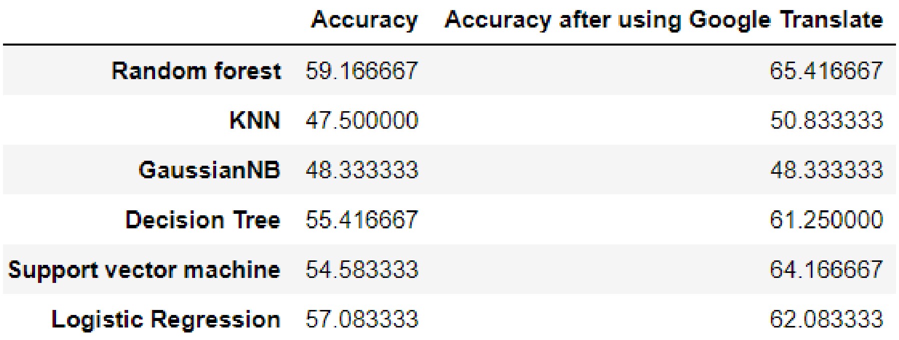
Thus, the implementation part can be concluded on obtaining the results of the machine learning algorithms. The outputs and results along with the statistical testing for the two obtained accuracies shall be discussed in the next section of ‘Result’.

Chapter 5: RESULT

In this section of the research, the outputs are displayed, and the results are compared.

Here, the outputs will be discussed for both the methods in the form of side by side comparison images and their details in a paragraph below. This is help understand the differences in the outputs and make it clear so as to which method was more effective in terms of accuracy and efficiency.

Before diving into the individual algorithm comparisons, the overall accuracy table for both the methods of sentiment analysis have been displayed below:



	Accuracy	Accuracy after using Google Translate
Random forest	59.166667	65.416667
KNN	47.500000	50.833333
GaussianNB	48.333333	48.333333
Decision Tree	55.416667	61.250000
Support vector machine	54.583333	64.166667
Logistic Regression	57.083333	62.083333

Figure 16: output table comparing accuracies of both methods of text analysis

From the table above, it can be seen that the google translated text data gave better accuracies than the original mixed language data. It helps us answer quite a few research questions while simultaneously raising new questions which shall be discussed in the Discussion section of the research. For now, let us move to the individual comparisons between algorithms.

The confusion matrix is a 3x3 matrix which shows how the algorithm has classified each class and how many were right, and how many wrong. The columns 1,2,3 are classes -1,0 and 1 respectively, while the rows 1,2,3 are also the classes -1,0 and 1 respectively. The values on the left (rows) show us the actual values while the values on the right(columns) show us the predicted values. Given below is a sample confusion matrix with explanation so as to how to read it.

	-1	0	1
-1	20	2	10
0	12	28	11
1	11	2	25

Here, cell (1,1) shows how many were actual -1 and how many were predicted -1. Cell 1x2 shows how many values were truly -1 but the algorithm predicted them to be 0. The cell 1x3 shows how many values were really -1 but were predicted to be 1. Similarly,

Cell (2,1): values were actually class 0 but were classified as -1. (wrong classification)

Cell (2,2): values were actually class 0 and were rightly classified as 0. (right classification)

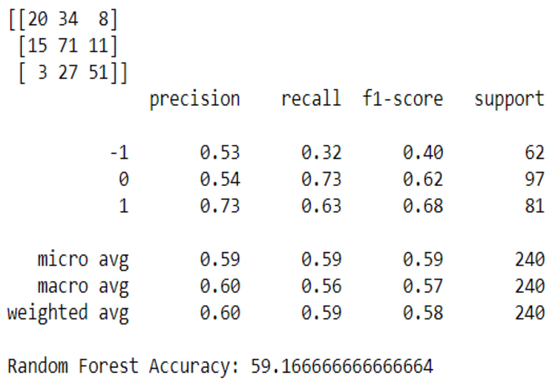
Cell (2,3): values were actually class 0 but were wrongly classified as 1. (wrong classification)

Cell (3,1): values were truly class 1 but were classified as -1. (wrong classification)

Cell (3,2): values were actually class 1 but were classified as 0. (wrong classification)

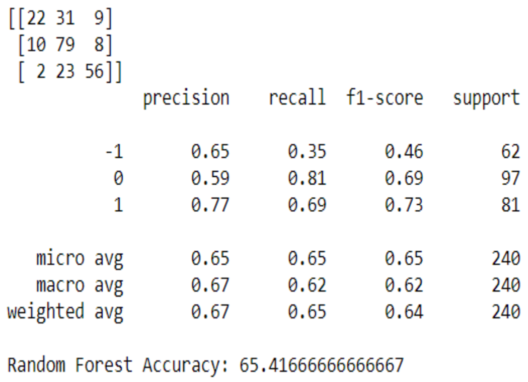
Cell (3,3): values were truly class 1 and were rightly classified as 1. (right classification)

5.1 Random Forest algorithm.



```
[[20 34  8]
 [15 71 11]
 [ 3 27 51]]
              precision    recall  f1-score   support

          -1       0.53      0.32      0.40        62
           0       0.54      0.73      0.62        97
           1       0.73      0.63      0.68        81

   micro avg       0.59      0.59      0.59       240
   macro avg       0.60      0.56      0.57       240
weighted avg       0.60      0.59      0.58       240

Random Forest Accuracy: 59.166666666666664
```

Figure 17: RF output or mixed language text

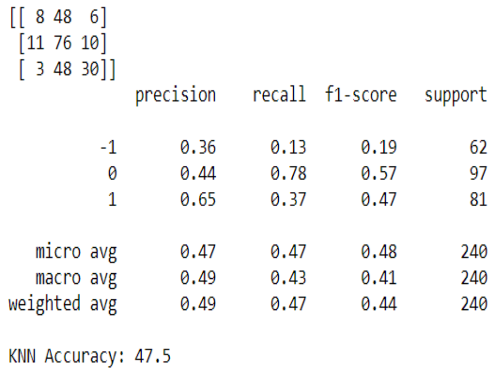


```
[[22 31  9]
 [10 79  8]
 [ 2 23 56]]
              precision    recall  f1-score   support

          -1       0.65      0.35      0.46        62
           0       0.59      0.81      0.69        97
           1       0.77      0.69      0.73        81

   micro avg       0.65      0.65      0.65       240
   macro avg       0.67      0.62      0.62       240
weighted avg       0.67      0.65      0.64       240

Random Forest Accuracy: 65.41666666666667
```

Figure 18: RF output for google translated text

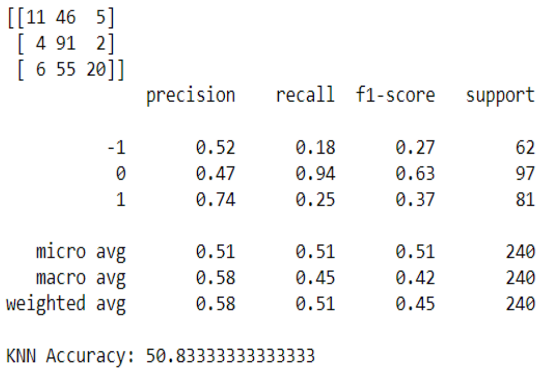
Here, we can see that the original text got an accuracy of 59.16% while the google translated text got an accuracy of 65.41%. The confusion matrix shows us that the google translated text showed much better performance in analysing the positive type sentiments than the original text. It also shows higher understanding of neutral sentiment and these two factors have helped increase the accuracy of the google translated text.

5.2. K Nearest Neighbour Algorithm.



```
[[ 8 48  6]
 [11 76 10]
 [ 3 48 30]]
              precision    recall  f1-score   support

          -1       0.36      0.13      0.19        62
           0       0.44      0.78      0.57        97
           1       0.65      0.37      0.47        81

   micro avg       0.47      0.47      0.48       240
   macro avg       0.49      0.43      0.41       240
weighted avg       0.49      0.47      0.44       240

KNN Accuracy: 47.5
```

Figure 19: KNN o/p for mixed language text

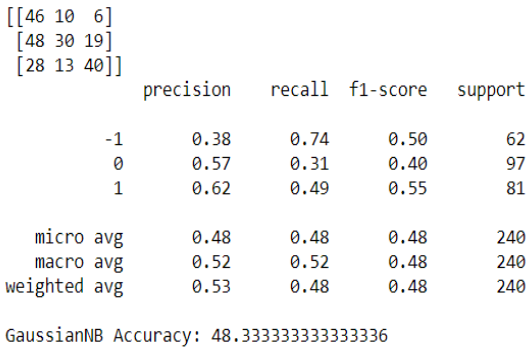


```
[[11 46  5]
 [ 4 91  2]
 [ 6 55 20]]
              precision    recall  f1-score   support

          -1       0.52      0.18      0.27        62
           0       0.47      0.94      0.63        97
           1       0.74      0.25      0.37        81

   micro avg       0.51      0.51      0.51       240
   macro avg       0.58      0.45      0.42       240
weighted avg       0.58      0.51      0.45       240

KNN Accuracy: 50.83333333333333
```

Figure 20: KNN o/p for google translated text

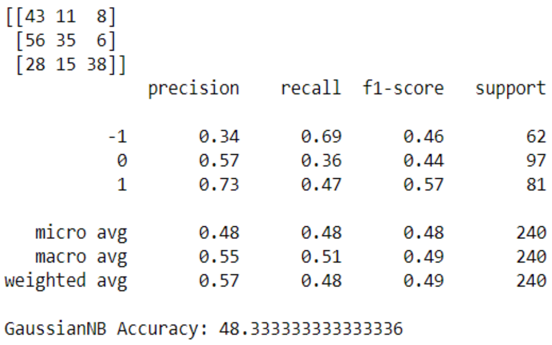
While the accuracies are quite close to each other, an increase in 3% is a big thing. It can be seen that Google translated KNN has succeeded in classifying neutral and negative sentences much more than the original text KNN. Earlier it was seen that some Google translated texts – on removal of stop words, were left with just one word. This might have played a big role in KNN being able to look for better neighbours than in the original case.

5.3. Naïve Bayes Algorithm.



```
[[46 10  6]
 [48 30 19]
 [28 13 40]]
              precision    recall  f1-score   support

          -1       0.38      0.74      0.50        62
           0       0.57      0.31      0.40        97
           1       0.62      0.49      0.55        81

   micro avg       0.48      0.48      0.48       240
   macro avg       0.52      0.52      0.48       240
weighted avg       0.53      0.48      0.48       240

GaussianNB Accuracy: 48.333333333333336
```

Figure 21: Naïve Bayes o/p for mixed language text

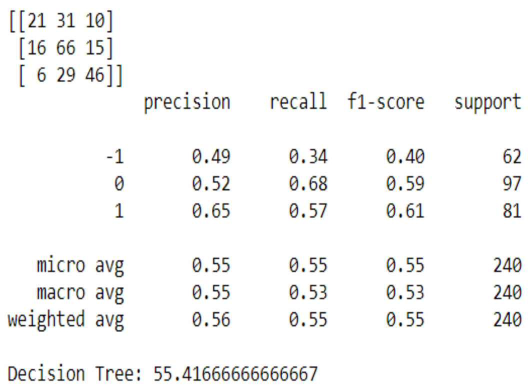


```
[[43 11  8]
 [56 35  6]
 [28 15 38]]
              precision    recall  f1-score   support

          -1       0.34      0.69      0.46        62
           0       0.57      0.36      0.44        97
           1       0.73      0.47      0.57        81

   micro avg       0.48      0.48      0.48       240
   macro avg       0.55      0.51      0.49       240
weighted avg       0.57      0.48      0.49       240

GaussianNB Accuracy: 48.333333333333336
```

Figure 22: Naïve Bayes o/p for google translated text

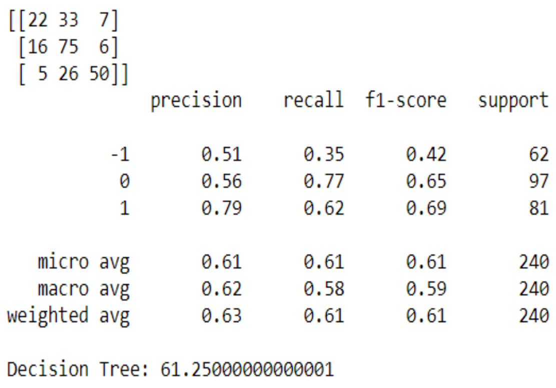
Surprisingly, it can be obsereved that at both the places, the accuracies are the same. The algorithm was excuted multiple times but each time same result was obtained. Sometimes such outputs do occur and it just means that the algorithm works same for both the processes.

5.4. Decision Tree Algorithm.



```
[[21 31 10]
 [16 66 15]
 [ 6 29 46]]
              precision    recall  f1-score   support

          -1       0.49      0.34      0.40        62
           0       0.52      0.68      0.59        97
           1       0.65      0.57      0.61        81

   micro avg       0.55      0.55      0.55       240
   macro avg       0.55      0.53      0.53       240
weighted avg       0.56      0.55      0.55       240

Decision Tree: 55.41666666666667
```

Figure 23: DT o/p for mixed language text

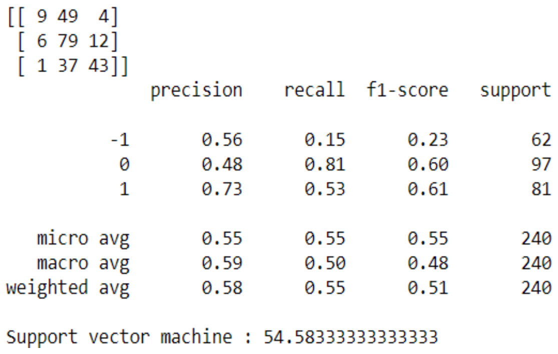


```
[[22 33  7]
 [16 75  6]
 [ 5 26 50]]
              precision    recall  f1-score   support

          -1       0.51      0.35      0.42        62
           0       0.56      0.77      0.65        97
           1       0.79      0.62      0.69        81

   micro avg       0.61      0.61      0.61       240
   macro avg       0.62      0.58      0.59       240
weighted avg       0.63      0.61      0.61       240

Decision Tree: 61.25000000000001
```

Figure 24: DT o/p for google translated text

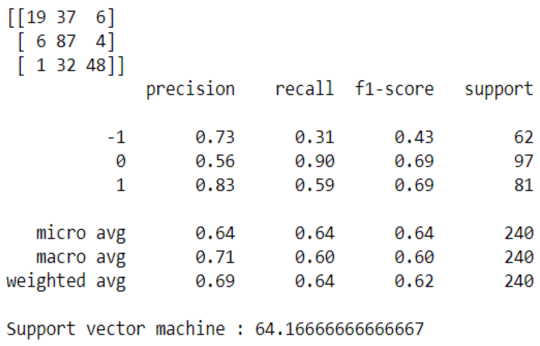
The output accuracies show a large difference here. While the original text gives a fairly average accuracy of 55.41%, the google translated text gives an accuracy of 61.25%. and the confusion matrix shows that apart from making more right classifications, Decision Tree has instead reduced the number of wrong predictions. Also, since Random Forest is basically a combination of a lot of decision trees, it was predictable that this algorithm will give less accuracy than Random Forest Algorithm.

5.5. Support Vector Machine Algorithm.



```
[[ 9 49  4]
 [ 6 79 12]
 [ 1 37 43]]
              precision    recall  f1-score   support

          -1       0.56      0.15      0.23        62
           0       0.48      0.81      0.60        97
           1       0.73      0.53      0.61        81

   micro avg       0.55      0.55      0.55       240
   macro avg       0.59      0.50      0.48       240
weighted avg       0.58      0.55      0.51       240

Support vector machine : 54.58333333333333
```

Figure 25: SVM o/p for mixed language text

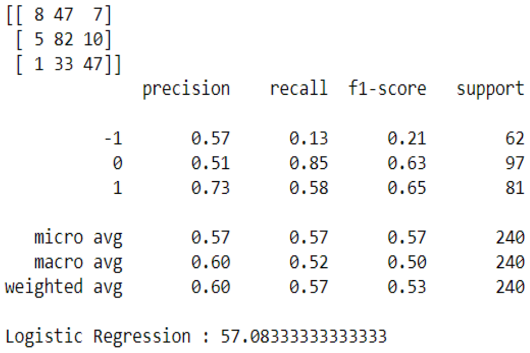


```
[[19 37  6]
 [ 6 87  4]
 [ 1 32 48]]
              precision    recall  f1-score   support

          -1       0.73      0.31      0.43        62
           0       0.56      0.90      0.69        97
           1       0.83      0.59      0.69        81

   micro avg       0.64      0.64      0.64       240
   macro avg       0.71      0.60      0.60       240
weighted avg       0.69      0.64      0.62       240

Support vector machine : 64.16666666666667
```

Figure 26: SVM o/p for google translated text

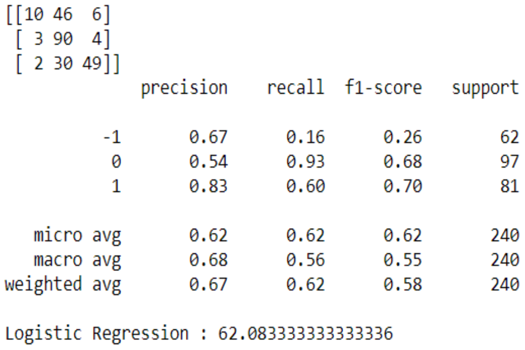
SVM shows the largest difference in accuracies for the classification. It can be seen in the confusion matrix that the google translated text gives very little errors while classifying the neutral sentences. The high output values in correctly predicting the sentences places SVM highly among other algorithms.

5.6. Logistic Regression Algorithm.



```
[[ 8 47  7]
 [ 5 82 10]
 [ 1 33 47]]
              precision    recall  f1-score   support

          -1       0.57      0.13      0.21        62
           0       0.51      0.85      0.63        97
           1       0.73      0.58      0.65        81

   micro avg       0.57      0.57      0.57       240
   macro avg       0.60      0.52      0.50       240
weighted avg       0.60      0.57      0.53       240

Logistic Regression : 57.08333333333333
```

Figure 27: LR o/p for mixed language text



```
[[10 46  6]
 [ 3 90  4]
 [ 2 30 49]]
              precision    recall  f1-score   support

          -1       0.67      0.16      0.26        62
           0       0.54      0.93      0.68        97
           1       0.83      0.60      0.70        81

   micro avg       0.62      0.62      0.62       240
   macro avg       0.68      0.56      0.55       240
weighted avg       0.67      0.62      0.58       240

Logistic Regression : 62.083333333333336
```

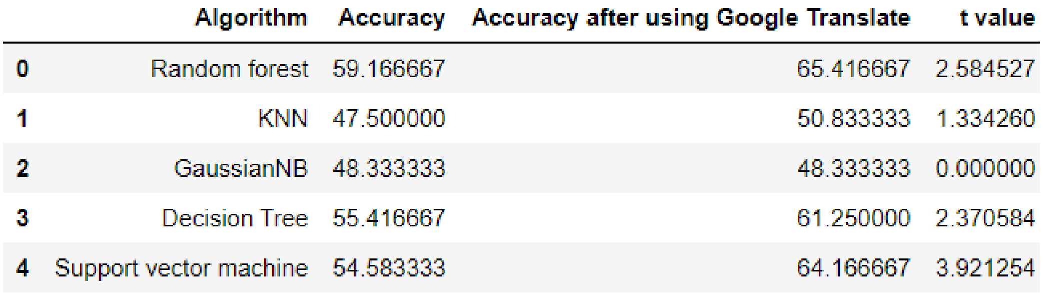
Figure 28: LR o/p for google translated text

Here, it can be observed that the neutral sentences have been reduced in errors by the logistic regression algorithm. The original text gave an accuracy of 57.08% whereas the google translated text gave an accuracy of 62.08%. It can be said that the improvement was noted due to the fact that while removing the stop words from the google translated texts, the reduced number of words allowed the algorithm to concisely apply the probabilistic classification prediction to the sentences.

5.7. Statistical testing:

Statistical testing is done in order to check the confidence level between the two accuracies compared.

After applying a t-test on the two accuracies obtained before and after translation, the following result was obtained:



	Algorithm	Accuracy	Accuracy after using Google Translate	t value
0	Random forest	59.166667	65.416667	2.584527
1	KNN	47.500000	50.833333	1.334260
2	GaussianNB	48.333333	48.333333	0.000000
3	Decision Tree	55.416667	61.250000	2.370584
4	Support vector machine	54.583333	64.166667	3.921254

Figure 29: t test output for the algorithms

T values and their meanings:

Value > 1.64 = Statistical difference at 90% confidence interval

Value > 1.96 = Statistical difference at 95% confidence interval

Value > 2.32 = Statistical difference at 99% confidence interval

As it can be seen, three out of the five algorithms show 99% confidence level. One shows a confidence level of 80+% and one shows zero.

Thus, it can be summed up by saying that the accuracies for the predictions range from the mid40s to the upper 60s. the highest performing algorithms were Random Forest Algorithm, Logistic Regression Algorithm, Decision Tree Algorithm and Support Vector Machine Algorithm.

Now that the results are obtained, the research questions and objectives all can be answered and explained. That shall be done in the next section of the research.

Chapter 6: CONCLUSION

The research focussed on obtaining mixed language text data and the same in Marathi text and then comparing to check how the accuracies while classification vary. Data was then taken, cleaned and put through text classification machine learning techniques which resulted in the following result.

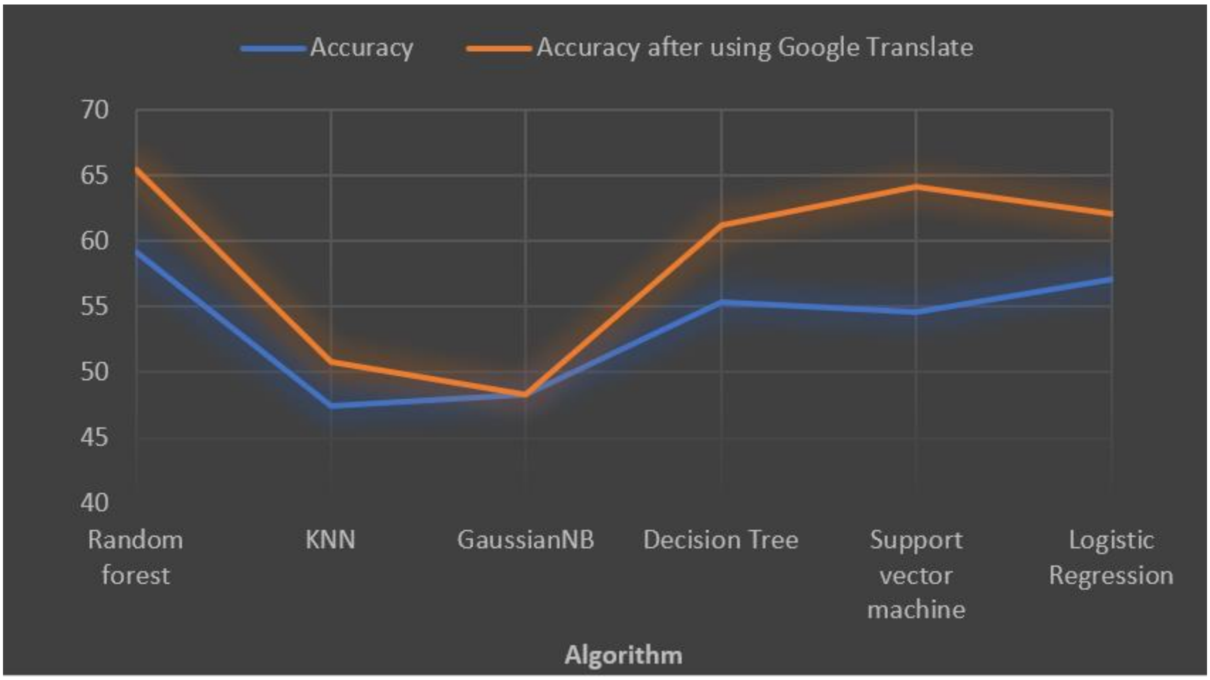


Figure 30: output for the comparison of the two text classification methods

From the table, the following things can be clearly noted:

The highest improvement or change in accuracy was noted by the SVM Algorithm – an improvement of almost 10%. This might have been possible due to the reasons that while removing the stop words from the google translated text, some sentences resulted in keeping just one or two words which probably helped the SVM Algorithm to create a better and clearer hyperplane and thus give better results on the translated text data.

The highest accuracy was shown by the Random Forrest Algorithm with 65.4%. This accuracy is higher than that of Decision Tree Algorithm even though both are similar algorithms, and this is because of the one major difference between the two algorithms. Random Forest is nothing but lots and lots of decision trees. This allows the algorithm to have different starting nodes and different attribute selections and this results in a much better result with higher accuracies. The surprising development was that Naïve Bayes gave the same output even after running the algorithm multiple times. The cause for this is unknown but .. (write more) KNN shows least change in accuracy with a rate of only 3%.

It can be concluded that a better change is seen when the data is translated and then classified rather than working on the original text. It also shows the google cloud translation's superior capability to translate words with a better accuracy. Also, it can be concluded that the objective defined in the introduction ae successfully fulfilled. And now the research questions can be answered too. The first research question demanded to know which algorithm gave the best result for the mixed language text. Form the table above it can be seen that the random forest algorithm gave the best output. The second research question demanded how the translated text performed on the same conditions. And it is evident from the result that the translated texts gave a much higher accuracy than the original texts. The third research question asked to compare the outputs, and that has been done extensively in the results section of the thesis.

The study also resulted in creation of an open source dataset for Marathi mixed language texts along with a dataset that provides the same sentences in typical Devanagari script. These can be considered as a contribution to the society and can be used by future data scientists to carry on and better the research.

6.1 Future Scope

It was mentioned in the introduction that the research, though done expensively on plain and single languages in their own script is available for most of the languages; the research for mixed languages or texts which are transliterated and code mixed is less. In this particular research, the importance of coming up with methods to correctly analyse the sentiment of the mixed language text has been stressed upon, for it is clear that in today's world maximum number of people are bilingual and type and text in different styles and texts. Hence improvements need to be done and the research should be focussed on correctly classifying these mixed texts.

The data for the research contained of sentences which are used on a day to day basis on different chat platforms or are spoken in a typical day. It was necessary to begin understanding and start implementing sentiment analysis to day to day talks for they are one of the most difficult areas to apply sentiment analysis and that too with high levels of accuracy. For that, this research can be considered a step in the right direction.

This particular research has implemented machine learning algorithms on the original mixed language texts and then found out the results. This can be bettered by methods like POS tagging and applying rule based language identification while using machine learning in order to obtain the right language of the dataset. POS tagging refers to parts of speech tagging and is usually considered to be a very difficult and unnecessary step to apply for the mixed language texts for the rules change on every step for the text. But with sufficient research, some form of POS tagging should be applicable.

Mixed language text for this research had two languages – English and Marathi. Sometimes, there might be three languages or more in a text. Studies can be done to try and implement sentiment analysis for that kind of datasets. Also, trying a lexicon based approach for the dataset can also be considered and the results compared. But the creation of a lexicon to near perfect conditions is not possible for the simple fact that every language has different dialects and compiling all of them might not be possible.

Also, as in all cases of languages, the base context upon which the statements were spoken may lead to a big difference in the actual sentiment of the sentences, hence even that could be developed upon. Sarcasm and irony are very difficult to decode and that may always be one of the reasons for not being able to obtain maximum efficiency.

PLAGIARISM AND REFERENCES

1. https://gulfnews.com/world/asia/india/census-more-than-19500-languages-spoken-in-india-as-mother-tongues-1.2244791 - news article
2. http://www.indiaonlinepages.com/population/population-of-maharashtra.html
3. https://cloud.google.com/translate/docs/reference/libraries/v3/python
4. E Mark Gold 1967, “Language Identification in the Limit”, Information and Control, Vol.10, Issue 5, pp. 447-474. https://doi.org/10.1016/S0019-9958(67)91165-5
5. Kamal Nigam, John Lafferty, Andrew McCallum (1999), “Using Maximum Entropy for Text Classification”, http://www.kamalnigam.com/papers/maxent-ijcaiws99.pdf
6. Andrea Esuli, Fabrizio Sebastiani, “SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining”, http://www.esuli.it/publications/LREC2006.pdf
7. Aditya Joshi, Balamurali A R, Pushpak Bhattacharyya, “A Fall-back Strategy for Sentiment Analysis in Hindi: a case Study”, https://www.cse.iitb.ac.in/~adityaj/HindiSentiWordnet_AdityaJ.pdf
8. Heba Elfardy, Mona Diab, “Token Level Identification of Linguistic Code Switching” https://www.aclweb.org/anthology/C12-2029.pdf
9. Akshat Bakliwal, Piyush Arora, Vasudeva Varma, “Hindi Subjective Lexicon: A Lexical Resource for Hindi Polarity Classification”, https://pdfs.semanticscholar.org/e7e6/e849600ab8ac499778c580c84d91616612b9.pdf
10. Bibekananda Kundu, Subhash Chandra, “Automatic Detection of English Words in Benglish Text”, IEEE International Conference on Intelligent Human Computer Interaction 2012, Kharagpur, India.

11. Yogarish Vyas, Spandana Gella, Jatin sharma, Kalika Bali, Monojit Choudhury, “POS Tagging of English-Hindi Code-Mixed Social Media Content”, Conference on Empirical Methods in Natural Language Processing (EMNLP) 2014, pp. 974-979. https://www.aclweb.org/anthology/D14-1105.pdf

12. Ben King, Steven Abney, “‘Labeling the Languages of Words in Mixed-Language Documents using Weakly Supervised Methods”, NAACL-HLT 2013, Georgia, pp.1110-1119. https://www.aclweb.org/anthology/N13-1131.pdf

13. Utsab Barman, Amitava Das, Joachim Wagner, Jennifer Foster, “Code Mixing: A Challenge for Language Identification in the Language of Social Media”, The First Workshop on Computational Approaches to Code Switching, Doha 2014, pp. 13-23. https://www.aclweb.org/anthology/W14-3902.pdf

14. Priyanka Pradip Kulkarni, Sonal Patil, Ganesh Dhanokar, “‘Marathi And Sanskrit Word Identification By Using Genetic Algorithm”, Intenatioanl Journal of Information & Futuristic Research 2015 (IJIFR), Vol 2, Issue 12, pp. 4588-4598. https://pdfs.semanticscholar.org/1029/43d392ac3ee33105642fbd8ddbc243e4ae31.pdf

15. Kumar Ravi, Vadlamani Ravi, “Sentiment Classification of Hinglish Text”, 3rd International Conference on Recent Advances in Information Technology (RAIT) 2016, DOI: 10.1109/RAIT.2016.7507974. https://www.researchgate.net/publication/305675500_Sentiment_classification_of_Hinglish_text

16. Muhammad Bilal, Huma Israr, Muhammad Shahid, Amin Khan, “Sentiment classification of Roman-Urdu opinions using Naive Bayesian, Decision Tree and KNN

classification techniques”. Journal of King Saud Univ. Computer and Information Sciences. 2016, 28, pp.330–344.

17. Harpreet Kaur, Dr. Veenu Mangat, Nidhi, “Dictionary based Sentiment Analysis of Hinglish text”, International Journal of Advanced Research in Computer Science (IJARCS), Vol.8, Issue 5, pp. 816-822, May-June 2017. https://pdfs.semanticscholar.org/812f/eb912829d8d4cd30d0a5a80bb935e6a1878b.pdf
18. Nafis Ietiza Tripto, Mohammed Eunus Ali, “Detecting Multilabel Sentiment and Emotions from Bangla Youtube Comments”, In Proceedings of the 2018 International Conference on Bangla Speech and Language Processing (ICBSLP), Sylhet, Bangladesh, 21–22 September 2018; pp. 1–6.
19. Sujata Deshmukh, Nileema Patil, Surabhi Rotiwar, Jason Nunes, “Sentiment Analysis of Marthi Language”, International Journal of Research Publications in Engineering and Technology (IJRPET) 2017, Vol 3, Issue 6, pp. 93-97. https://journalnx.com/papers/20150361-sentiment-analysis.PDF
20. Mohammed Arshad, Sharvari Govilkar, “Sentiment Analysis of Mixed Code for the Transliterated Hindi and Marathi Texts”, International Journal o Natural Language Computing (IJNLC) 2018, Vol 7, Issue 2, pp. 15-28. DOI: 10.5121/ijnlc.2018.7202. http://aircconline.com/ijnlc/V7N2/7218ijnlc02.pdf
21. Abhishek Kaushik, Gagandeep Kaur, Shubham Sharma, “Cooking is creating emotion: A study on Hinglish sentiments of Youtube cookery channels using semi supervised approach”, Big Data and Cognitive Computing, July 2019, pp. 4-19.

APPENDIX

This Section contains the information on what the artefacts folder contains and what all the files mean.

The folder of ‘thesis artefacts’ submitted along with this particular research document contains the following files:

Datasets:

1. marathi data.csv : a csv file that contains the marathi dataset along with the rating for the same. Can be opened in MS Excel.

2. results.csv : a csv file that shows the output for the research along with the t-score. Can be opened in MS Excel.

3. tdata.csv : a csv file that shows the mixed language text dataset used in the research along with the rating for the same. Can be oepend in MS Excel.

4. ttext.csv : a csv file that shows the marathi sentences and their English translations obtained from google translate. Can be opened using MS Excel.

5. thesis dataset.csv: acsv file that shows the original dataset where all 8 participants had rated the dataset.

Stop word files:

1. english-new: a file that contains the manually constructed English stop word list which can be opened in a simple notepad.

2. marathi: a file that contains the manually written Marathi stop words list which can be opened in a simple notepad.

Python Files:

1. Thesis python code.ipynb : this file contains the code for the entire research and is in the '.ipynb' format. It can be opened using jupyter notebook.