Text Emotion Detection Using Machine Learning Algorithms

Vishakha Singh
Computer Engineering Department,
V.E.S. Institute of Technology,
Chembur, Mumbai, India
2021.vishakha.singh@ves.ac.in

Manasi Sharma
Computer Engineering Department,
V.E.S. Institute of Technology,
Chembur, Mumbai, India
2021.manasi.sharma@ves.ac.in

Anushka Shirode
Computer Engineering Department,
V.E.S. Institute of Technology,
Chembur, Mumbai, India
2021.anushka.shirode@ves.ac.in

Prof. Sanjay Mirchandani
Computer Engineering Department,
V.E.S. Institute of Technology,
Chembur, Mumbai, India
sanjay.mirchandani@ves.ac.in

***Abstract*—Text Emotion determination is becoming more vital with every passing day as millions of text messages are released on a daily basis. In text analysis, the challenge lies in interpreting the ambiguity of human language, particularly when it comes to emotions. A study was carried out to detect the emotions classified into six categories as anger, fear, joy, love, sadness, and surprise. The algorithms, including Logistic Regression, Linear Support Vector Machine, and Random Forest were used for detecting and classifying the emotions. A comparative study of these methods was carried out by considering two features namely Term Frequency-Inverse Document Frequency and Count Vectors. The highest accuracy was obtained for the Count Vectors feature using the Logistic Regression method and the emotions in the text were classified accordingly.**

Keywords—Datasets, Accuracy, Analysis.

I. Introduction

Emotions are psychological sentiments, and understanding a person's emotions through text is challenging as the actual emotion depends on an accurate interpretation of the text. The accuracy of identifying others' emotions varies depending on the individual. Emotion recognition technology is an emerging research area [1].

The algorithms chosen to carry out the study were Logistic Regression, Linear Support Vector Machine (Linear SVM), and Random Forest which are supervised machine learning algorithms as they work with labeled datasets. Logistic Regression is a simple and interpretable model. Linear SVM works well with high-dimensional data and the Random Forest algorithm is less prone to overfitting and captures both linear and non-linear relationships.

The goal of using the algorithms mentioned is to accurately determine the emotions expressed in the text, such as anger, sadness, love, joy, fear, and surprise.

II. Literature review

The research papers and surveys referred to include:

A. "Emotion Detection and Recognition from Text using Machine Learning" by Shaikh Abdul Salam and Rajkumar Gupta [2].

The algorithms used in this paper are Support Vector Machine (SVM), K-Means, and Naive Bayes.

The merits of this paper include the use of multiple classifiers, sentence-level analysis, and multiple emotion classes. The demerits observed were the need to improve pre-processing data, further implementation of semantic-level parsing, and inclusion of implicit emotions in objective sentences in future work.

B. "Affective Computing And Sensing Analysis," a survey on sentiment analysis algorithms and applications by Waala Medhat, Ahmed Hassan, and Hoda Korashy, uses SVM and Maximum Entropy [3].

The merits include a comprehensive overview and classification of lexicon-based, machine learning-based, and hybrid approaches. The demerits observed were a lack of discussion of ethical issues and limited discussion of feature selection and evaluation metrics.

C. In "Emotion Detection from Text," by Shiv Naresh Shivhare and Prof. Saritha Khethawat [4].

They used a self-proposed algorithm called Emotion Word Ontology which uses a hit-or-miss approach.

The merits include the use of multiple emotion classes and chi-square feature selection. The demerits were limited sample size and limited discussion of real-world applications.

III. Methodology

A. Logistic Regression

Logistic Regression analysis is used to predict a binary outcome (0 or 1) based on input variables. The probability of the dependent variable being in a certain class is based on the input variables. It uses a sigmoid function [6] to transform the output of linear regression into a value whose probability lies between 0 and 1.

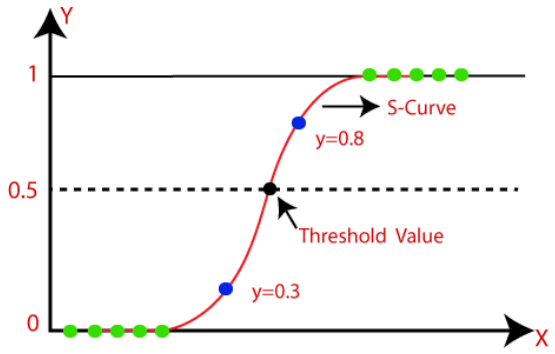

Fig. 1. Logistic Regression S-Curve [5].

The sigmoid function predicts the output of the dependent variable. It is an S-shaped graph. If the output probability is above the threshold value it will be mapped to one, and if less than the threshold it will be mapped to zero as shown in Fig. 1.

Logistic regression is an illustrative and simple algorithm that works well with linearly separable data.

B. Linear Support Vector Machine

The aim is to find an appropriate hyperplane that maximizes the margin between classes, which can be useful when there is a clear boundary between two classes. It works well with high-dimensional data and linearly separable data.

It is a supervised machine learning algorithm used for classification as well as regression problems [6].

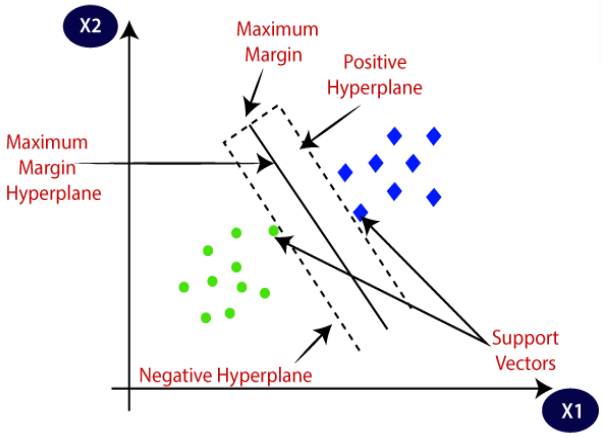

Fig. 2. Linear SVM Classifier [7].

Linear SVM has the ability to handle large datasets efficiently. This is because the optimization problem that linear SVM solves can be formulated as a quadratic programming problem, which can be efficiently solved using various optimization techniques. This means that linear SVM is able to create a decision boundary that is optimal in terms of separating the different classes in the dataset.

The final prediction of the emotion is done based on the distance of the data points from this hyperplane.

B. Random Forest Algorithm

Random Forest is an ensemble learning algorithm that uses bagging. It selects subsets of data and features randomly and builds decision trees on these subsets. It has the ability to handle high-dimensional data.

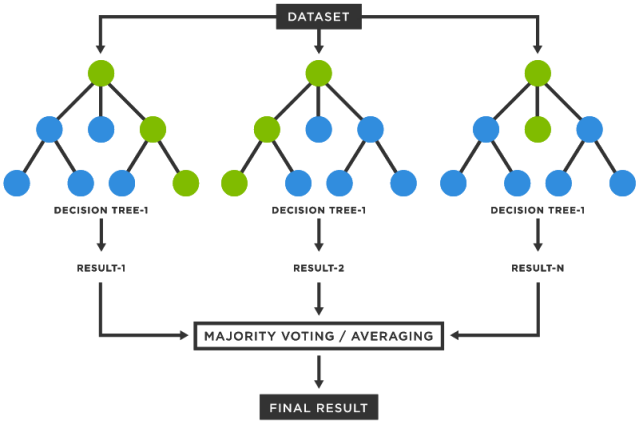

Fig. 3. Random Forest Representation [8].

It can automatically select the most important features in the dataset, which can help to simplify the model and improve its performance.

It is less prone to overfitting than individual decision trees.

One of the most important features of the Random Forest Algorithm is that it can handle the data set containing continuous variables as in the case of regression and categorical variables as in the case of classification. It performs better results for classification problems.

The dataset is divided into decision trees as shown in Fig. 3. that predicts the individual results. This is called bootstrapping. These results are then aggregated using majority voting in order to predict the final result.

IV. Implementation

The six stages involved in the functional working of the model as shown in Fig. 4. are as follows:

Stage 1: Setting up the Development Environment.
The IDE used is 'Google Colab'.

Stage 2: Choosing dataset.
The Pandas library of Python has been used to read the dataset.

Stage 3: Understanding the dataset.
The size of the dataset is 16,000 sentences. The text file contains columns for the sentences and the corresponding emotion it depicts.

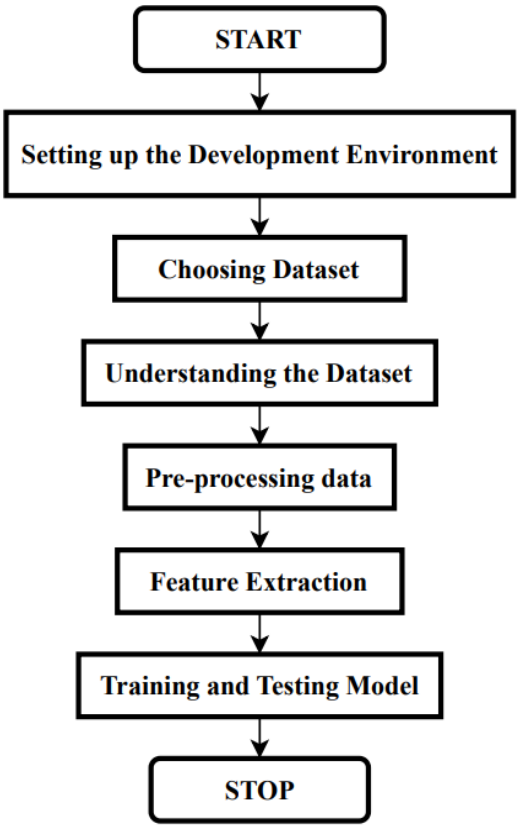

Fig. 4. Working of the model.

Stage 4: Pre-processing data:
For uniformity, the text is converted to lowercase. Irrelevant words like stop words and punctuation marks have been removed. Lemmatization, i.e., converting all words to their root form, has been performed [9][10]. NLTK and re-module have been used for the same.

Stage 5: Feature Extraction:
The data is error-free and clean. For feature extraction, Term Frequency-Inverse Document Frequency (TF-IDF) and Count Vectors features have been considered [11]. The TfidfVectorizer and CountVectorizer of the sklearn module have been used for the same. Both these feature extraction techniques are used to convert raw text data into a numerical representation that can be fed to machine learning algorithms.
TF-IDF measures the relative importance of a word in the data. TF-IDF increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus.
Count Vectors are a simple way to represent text data as a matrix of token counts. Each row represents a document from the corpus, and each column represents the count of a specific token in that document.

Stage 6: Training and Testing the Model:
The dataset has been split into three ratios namely 80:20, 75:25, and 70:30 for training and testing, respectively. The accuracy of the model has been enhanced by setting hyperparameters like max_iter and n_estimators. The max_iter parameter mentions the maximum number of iterations that have to be performed on the training data. N_estimators used in the random forest model mention the number of trees.

V. Results

A. Confusion Matrix

	Predicted 0	Predicted 1	Predicted 2	Predicted 3	Predicted 4	Predicted 5
Actual 0	636	14	33	3	44	0
Actual 1	21	545	18	2	37	25
Actual 2	9	10	1706	56	26	10
Actual 3	3	2	97	328	9	0
Actual 4	38	22	48	6	1456	4
Actual 5	1	27	19	1	8	136

Fig. 5. Confusion Matrix.

Fig. 5. shows a confusion matrix, which is a performance measure of a classification algorithm [12]. The matrix is a table used to evaluate the algorithm's performance by summarizing the number of correct and incorrect predictions and identifying the types of errors made. This helps in determining the algorithm's effectiveness and identifying areas that require improvement. The matrix also helps in understanding the true positives and negatives, with the diagonal values representing the accurate predictions.

B. Heatmap

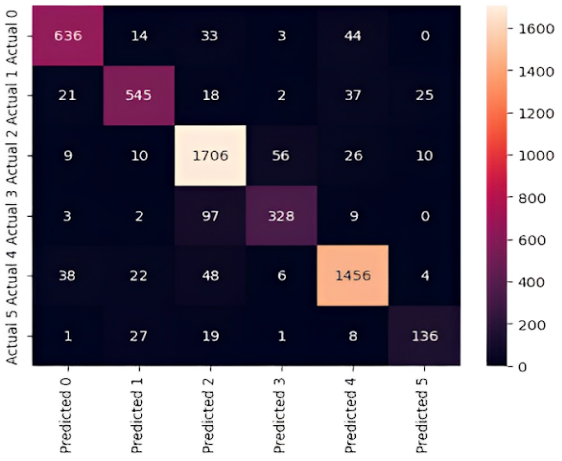

Fig. 6. Heatmap.

Fig. 6. displays a heat map, which is a graphical representation of the confusion matrix.

A heat map is a graphical depiction of data in which values are illustrated using colors. The values are generally presented as a matrix or a table, and the colors reflect the intensity of the values. Heat maps are employed to identify patterns or trends in large datasets, making it easier to analyze and interpret the data. Heat maps are not restricted to two-dimensional representations - they can also be utilized to visualize data in three dimensions, where colors signify the intensity of values in the z-axis.

C. *Output Obtained*

The following are the outputs obtained from user inputs:

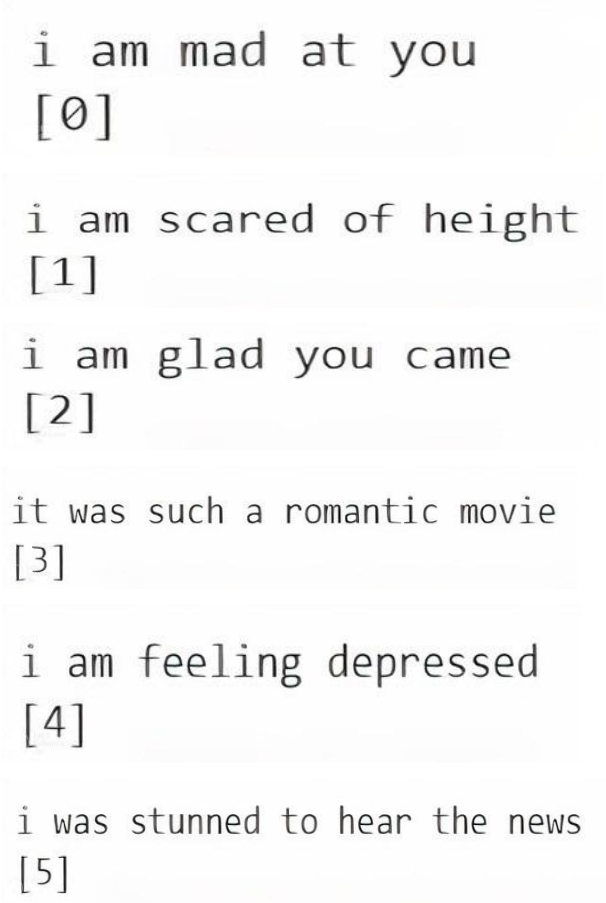

Fig. 7. Output.

The output in Fig. 7. uses the following mapping for the emotions [13][14]:

- 0 indicates anger.
- 1 indicates fear.
- 2 indicates joy.
- 3 indicates love.
- 4 indicates sadness.
- 5 indicates surprise.

The input sentences are provided by the user, and the model predicts the corresponding emotions.

D. *Accuracy Chart with Data Split Ratio*

Table I. Comparison of accuracies obtained

Features	Dataset Split Ratio	Accuracy Obtained		
		Linear SVM	Random Forest	Logistic Regression
TF-IDF	80:20	29.86%	28.41%	32.16%
	75:25	27.73%	28.15%	29.84%
	70:30	31.2%	30.11%	33.03%
Count Vectors	80:20	88.30%	86.13%	88.83%
	75:25	88.26%	86.02%	88.55%
	70:30	88.35%	86.12%	89.01%

Table I shows the final accuracies obtained from the features TF-IDF and Count Vectors using Linear SVM, Random Forest, and Logistic Regression methods respectively. Better accuracies were obtained by considering a dataset split ratio of 70:30 for the feature TF-IDF as 31.20% for Linear SVM, 30.11% for Random Forest, and 33.01% for Logistic Regression. On the other hand, the accuracies obtained using Count Vectors were 88.35% for Linear SVM, 86.12% for Random Forest, and 89.01% for Logistic Regression.

As the frequency is lower in TF-IDF, its accuracy is lower than that of Count Vectors.

The bar graphs [15] in Fig. 8,9, and 10 display a comparison of the accuracies obtained for each algorithm using various data split ratios using TF-IDF and Count Vectors features.

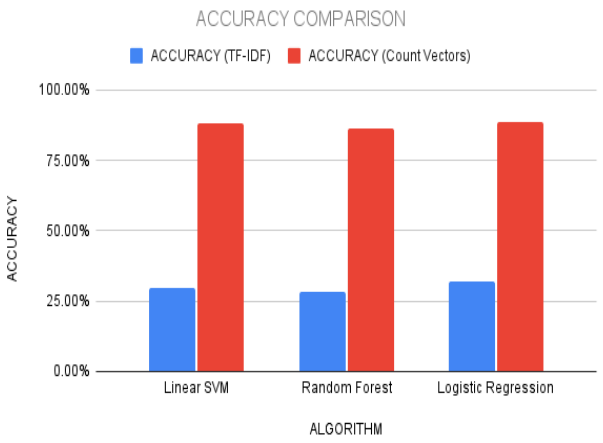

Fig. 8. Accuracy Chart for dataset split ratio of 80:20.

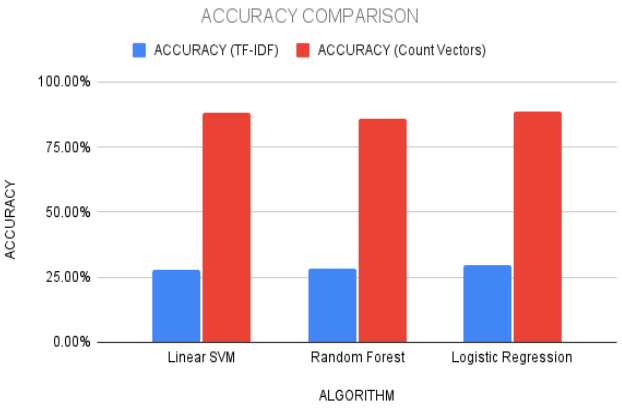

Fig. 9. Accuracy Chart for dataset split ratio of 75:25.

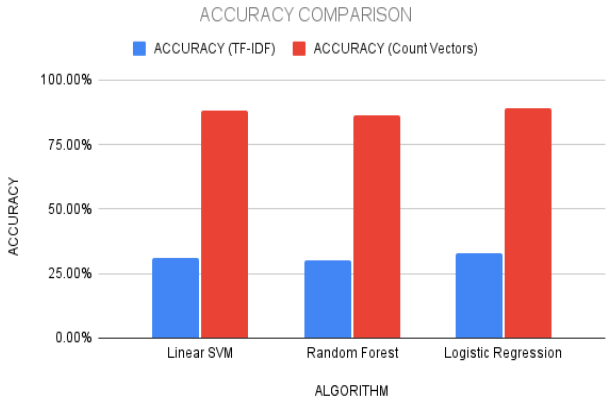

Fig. 10. Accuracy Chart for dataset split ratio of 70:30.

E. Observations

Out of the three splitting ratios of the training and testing dataset, that is 80:20, 75:25, and 70:30, the highest accuracy was obtained by considering the dataset split ratio as 70:30.

It was also observed that the accuracy of the model increased by incrementing the max_iter parameter.

VI. Conclusion And Future Scope

The study determined that the highest accuracy of 89.01% was achieved for the Count Vectors feature using the Logistic Regression method by considering a dataset split ratio of 70:30. The six emotions were classified according to the input given by the user.

In conclusion, with the increasing volume of text data being generated every day, accurately classifying emotions from the text can be incredibly valuable, and more emotions can be considered for classification. Further study of additional potential classifiers or ensemble techniques, such as K-Means and the Naive Bayes Classifier can be explored to improve the results.

References

[1] F.A. Acheampong, F. Adoma, W. Chen, N.-M.Henry, *Text-based emotion detection: Advances, Challenges, and Opportunities. Eng. Rep. 2(7), e12189 (2020).*

[2] Salam, Shaikh & Gupta, Rajkumar. (2018). Emotion Detection and Recognition from Text using Machine Learning. International Journal of Computer Sciences and Engineering. 6. 341-345. 10.26438/ijcse/v6i6.341345.

[3] Walaa Medhat, Ahmed Hassan, Hoda Korashy, *Ain Shams Engineering Journal*, Volume 5, Issue 4, December 2014, Pages 1093-1113. Sentiment analysis algorithms and applications.

[4] Shiv Naresh & Khethawat, Saritha. (2012). Emotion Detection from Text. Computer Science & Information Technology. 2. 10.5121/csit.2012.2237

[5] https://www.javatpoint.com/logistic-regression-in-machine-learning

[6] https://intellipaat.com/blog/tutorial/machine-learning-tutorial/svm-algorithm-in-python/

[7] https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm

[8] https://www.tibco.com/reference-center/what-is-a-random-forest

[9] Sun, Shiliang; Luo, Chen; Chen, Junyu (July 2017). "*A review of natural language processing techniques for opinion mining systems*". Information Fusion. 36: 10–25. Doi: 10.1016/j.inffus.2016.10.004.

[10] https://www.javatpoint.com/data-preprocessing-machine-learning

[11] https://enjoymachinelearning.com/blog/countvectorizer-vs-tfidfvectorizer/

[12] https://www.simplilearn.com/tutorials/machine-learning-tutorial/confusion-matrix-machine-learning

[13] https://www.textrics.ai/solutions/emotion-detection

[14] https://www.sciencedirect.com/science/article/pii/S2772528622000103#:~:text=Emotion%20extraction%20from%20the%20text,data%20mining%20and%20so%20on

[15] https://web.mit.edu/course/21/21.guide/grf-bar.htm