2023 International Conference on Computational Intelligence for Information, Security and Communication Applications (CIISCA) | 979-8-3503-3972-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/CIISCA59740.2023.00056

Sentiment Analysis of Twitter using Machine learning algorithms

Rajath N
Dept. of Artificial Intelligence & Data Science
Global Academy of Technology
Bangalore, India
rajathrajns435@gmail.com

Kushal V
Dept. of Artificial Intelligence & Data Science
Global Academy of Technology
Bangalore, India
kushalgowda056@gmail.com

Ashwini Kodipalli
Dept. of Artificial Intelligence & Data Science
Global Academy of Technology
Bangalore, India
ashwini.kodipalli@gmail.com

Trupthi Rao
Dept. of Artificial Intelligence & Data Science
Global Academy of Technology
Bangalore,India
trupthirao@gat.ac.in

Pushpalatha V.
Dept. of Artificial Intelligence & Data Science
Global Academy of Technology
Bangalore,India
pushpav27@gmail.com

Rohini B.R.
Dept. of Artificial Intelligence & Data Science
Global Academy of Technology
Bangalore,India
rohini.br@gmail.com

***Abstract*—Twitter sentiment analysis has numerous applications in social media monitoring, brand management, customer service, political analysis, and more. By analysing the sentiment of tweets, businesses can gain insights into customer behaviour, improve customer engagement, and enhance brand reputation. In the following paper, based on many factors including id, location, target, and text, we suggest using machine learning as a method to examine the sentiment of a number of tweets. We use different algorithms such as Random Forest, Naive Bayes, logistic regression and K-nearest neighbours, Gradient Boosting Classifier, Support Vector Machine(SVM), Decision Tree Classifier, Bagging Classifier, Ada Boosting classifier, gradient boosting classifier to train and test our model on a dataset of 1000 tweets with positive and negative sentiments. A comparison of the classifiers' performance is made using measures for accuracy, precision, recall, and F1-score. We find that logistic regression and SVM algorithms achieve the highest of the accuracy which is 58% and F1-score of 0.62 for logistic regression and 0.57 for SVM. We come to the conclusion that our machine learning strategy can offer a trustworthy and effective tool for sentiment analysis of tweets.**

Keywords—Sentiment Analysis, Machine Learning, SVM, Logistic regression

I. INTRODUCTION

Sentiment analysis of twitter is a potent technique that enables us to comprehend the views, attitudes, and feelings people on Twitter have towards a specific subject, good, or event. As social media usage has increased massively and daily data production has skyrocketed, sentiment analysis has emerged as a crucial tool for businesses, organisations, and researchers to examine and understand how the general public views their brand, product, or service. Sentiment analysis involves categorising tweets as positive tweet, negative tweet, or neutral based on the sentiment indicated in the form of text using NLP and ML techniques. We can spot patterns, monitor shifts in public opinion, and obtain important insights into consumer behaviour by analysing the sentiment of tweets. Making data-driven decisions, increasing customer engagement, and boosting brand reputation can all be done with the use of this information. This research paper is organized in the following manner. Section_2 lists the literature survey, Section_3 describes the methodology, Section_4 represents the analysis of results and the article is concluded in Section_5.

II. LITERATURE SURVEY

In recent years, machine learning-based sentiment analysis of tweets has been a hot topic of study. In order to create efficient models for sentiment analysis on Twitter data, several experiments have been carried out. We shall examine some of the major conclusions and contributions of these studies in this literature review.

Pak and Paroubek (2010) [1][2][3] carried out one of the earliest works in this field in which they used a supervised learning strategy for sentiment analysis on Twitter data. They divided tweets into three categories—positive, negative, and neutral—using a Nave Bayes classifier. They attained a 60% accuracy for the three-class classification and an 80% accuracy for binary classification.

Semi-supervised learning was employed in a different work by Go et al. (2009) [4][5][6][7] to analyse sentiment in Twitter data. To categorise tweets as favourable or negative, they combined lexical resources and machine learning approaches. They were 86.4% accurate in their binary classification.

For sentiment analysis on Twitter data, Wang et al. (2012) [8][9][10][11] combined machine learning and rulebased methods. They divided tweets into four groups using a Maximum Entropy classifier and a set of rules: objective, neutral, negative, and positive. They were 69.75% accurate in their four-class classification.

Zhang et al. (2021) [12][13][14][15] recently suggested a hybrid model for sentiment analysis on Twitter data that incorporates deep learning and machine learning approaches. After extracting information from tweets with a convolutional neural network (CNN), they classified tweets into three categories: positive, negative, and neutral using a support vector machine (SVM) classifier. For three-class categorization, they attained an accuracy of 89.92%.

A unique method for sentiment analysis on Twitter data employing a graph convolutional neural network (GCN) was suggested in a recent article by Chen et al. (2021) [16][17][18][`9]. After gathering contextual data from a graph-based representation of tweets, they utilised GCN to categorise tweets into three groups: good, negative, and neutral. A three-class classification accuracy of 86.44% was attained. [20][21][22][23].

(1 -positive, -1 -negative,0 -neutral)

979-8-3503-3972-7/23/$31.00 ©2023 IEEE
DOI 10.1109/CIISCA59740.2023.00056

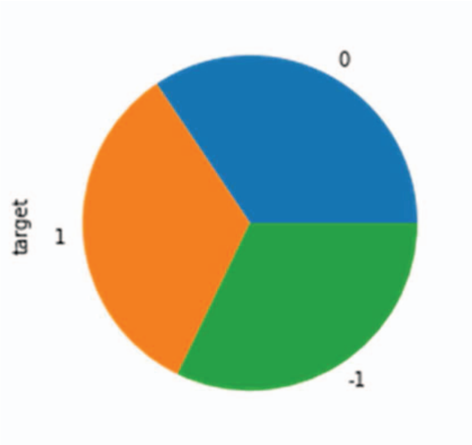

Fig.1.Three Class Classification Accuracy

In summary, the papers examined in this literature review show that using machine learning approaches, tweets can be accurately categorised into three categories: positive tweet, negative tweet, and neutral for sentiment analysis on Twitter data. These studies emphasise the value of hybrid models, feature engineering, and contextual data in achieving high accuracy for sentiment analysis on Twitter data.

III. METHODOLOGY

A. Data Description

The following dataset used for this research is taken from Kaggle. The entity level is used in this dataset for sentiment analysis on Twitter. The goal is to evaluate the mood of the message regarding the entity when given a message and an entity. This dataset has three classes: Positive, Negative, and Neutral. We view messages that are Irrelevant as Neutral.

The dataset includes the following attributes: -

Tweet ID: - Unique identifier

Entity: - Type of tweet

Sentiment: - (positive, negative, neutral)

Tweet Content: -Tweets

B. Dataset Analysis

With 30% of the data in the test set and 70% of the data in the training set, the dataset is divided into these two sets: training and test. To analyse the sentiments of tweets, the data is then used to train classification algorithms. For in-depth data collection, visual representation in the form of bar graphs and pie charts is created.

C. Architecture Framework

The implementation of this research includes opensource libraries such as Numpy, Pandas, Matplotlib and Seaborn for complex mathematical computation of values, graphical representations and dataset analysis respectively to arrive at meaningful analytics and predict accurate outcomes.

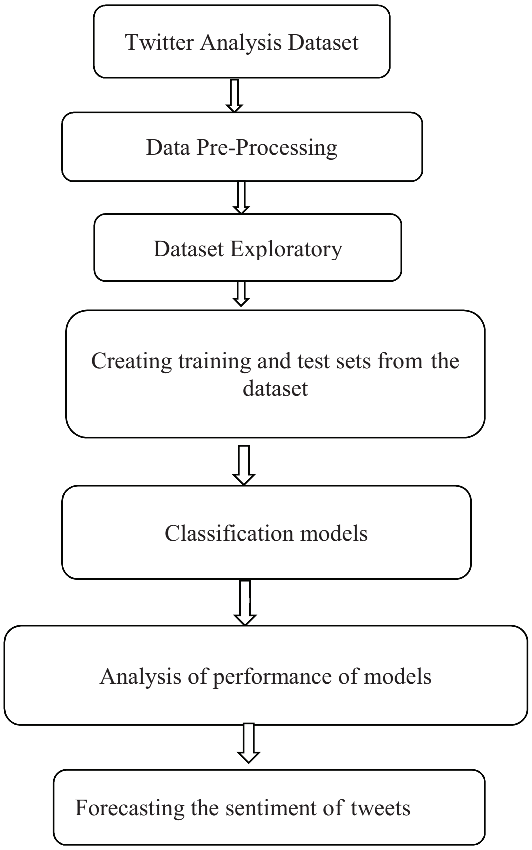
Twitter Analysis Dataset
Data Pre-Processing
Dataset Exploratory
Creating training and test sets from the dataset
Classification models
Analysis of performance of models
Forecasting the sentiment of tweets

FIG.2.Architecture for Sentiment Analysis in Twitter

D. Classification Models

Logistic regression: Is a statistical technique used to examine the connection between a dependent variable and one or more independent variables. The dependent variable is often binary, taking one of two potential values. It is frequently applied to binary classification issues in the field of machine learning.

Logistic regression seeks to identify the most precise model that can predict the Twitter Analysis Dataset Data Pre-processing Dataset Exploratory Analysis for all features creating training and test sets from the dataset Building various classification models such as Logistic Regression, Naïve Bayes, KNN, SVM, Gradient Boosting, Ada Boosting, Random Forest, Decision Tree Analysis of performance of models with metrics such as accuracy, f1-score, specificity, support Forecasting the sentiment of tweets probability of the dependent variable based on the independent variables. The logistic regression model, also known as a sigmoid function, uses a logistic function, often referred to as a logistic function, to predict the likelihood that the dependent variable would take a certain value given the values of the independent variable. orCGSaprimary unit The logistic function can be defined as:

$f(x)=1/(1+e^{-x})$

where x is the linear combination of the independent variables and their coefficients.

The popularity of logistic regression can be attributed to its relative simplicity, speed, and readability. It is vulnerable to outliers and multicollinearity, though it makes the assumption that the relationship between the independent variables and the dependent variable is linear.

Gaussian Naive Bayes: Is a variant of the Naive Bayes algorithm used for classification. It assumes that the features in the dataset are normally distributed and independent of each other.

The Bayes theorem, which says that the probability of a hypothesis (in this case, a class label) is proportional to the probability of the data given the hypothesis times the prior probability of the hypothesis, is used by the algorithm to determine the probabilities of each class label given the input features. In Gaussian Naive Bayes, it is expected that the probability of the data given the class label will follow a Gaussian (normal) distribution. In order to determine the likelihood of each class label given the input features, the mean and variance of each feature for each class are estimated using the training data.

The advantage of Gaussian Naive Bayes is that it is computationally efficient and can be trained on relatively small datasets. However, it can perform poorly if the assumption of normality is violated or if there is strong dependence between the features.

K-nearest neighbour (KNN): An efficient and straightforward approach that is applied to both regression and classification. It belongs to the class of instance-based learning, where the algorithm makes predictions by comparing the input data with the labeled data points in training set.

In KNN, the "K" refers to number of nearest neighbours to consider for classification or regression. For example, if K is set to 5, then the algorithm will consider the five closest labeled data points to the input and classify or predict upon the majority label or average of those neighbours.

To determine the distance between the input data and the labelled data points, Euclidean distance or other distance metrics are frequently utilised. The KNN algorithm's performance can be greatly impacted by the distance measure chosen and the value of ‘K’.

One advantage of KNN is that it is a nonparametric technique and does not presuppose any information about how the data is distributed. On the other hand, it struggles in feature spaces with high dimensionality and can be computationally expensive for large datasets. Furthermore, the selection of the distance metric and the value of K, which may need to be adjusted through cross-validation, can have an impact on how well a KNN performs.

Support Vector Machine (SVM): is a well-liked supervised learning method used for outlier detection, regression, and classification. In order to maximise the margin between several classes of data points, SVM seeks out the best hyperplane in a high-dimensional feature space..

In SVM, the algorithm looks for a hyperplane that, with the greatest feasible margin, the data points are divided into distinct classes in the feature space. Support vectors are the nearest data points to the hyperplane, and they are very important in establishing the location and orientation of the hyperplane.

By utilising various kernel types, SVM can able to handle data that can be separated into linear and non-linear categories. A kernel is a function which raises the data's spatial dimension, making it separable by a hyperplane.

The linear, polynomial, and radial basis function (RBF) are some examples of frequently used kernels.

One of the advantages of SVM is that it can be effective in high-dimensional spaces, even with a limited number of training samples. Additionally, SVM is less susceptible to overfitting than other algorithms and can handle non-linearly separable data using different kernel functions. However, SVM can be sensitive to the choice of kernel and its hyperparameters, which may need to be tuned through cross-validation. SVM is also computationally intensive, particularly when dealing with large datasets.

Gradient Boosting Classifier (GBC): Belonging to the family of ensemble learning algorithms and is a well-known machine learning method. To provide a final forecast that is more accurate, ensemble learning combines the predictions of several different separate models. Weak decision trees are utilised in GBC as separate models, and the combined forecasts from these models form the final prediction.

The algorithm works by iteratively fitting new decision trees to the residuals (difference between predicted and actual values) of the previous decision tree. Each subsequent tree is trained to predict the remaining error or residuals of the previous tree. This process is repeated until the residuals can no longer be improved or until a maximum number of trees is reached.

In GBC, each decision tree is typically shallow, and its nodes are split using a greedy algorithm based on maximizing the information gain or reducing the impurity of the target variable. The difference between the predicted and actual values is measured by the loss function, which is minimised by GBC using a gradient descent optimisation process.

AdaBoost (Adaptive Boosting) Classifier: Belonging to the family of ensemble learning algorithms and is a well-known machine learning method. It creates a strong classifier by combining several weak ones. A model that only slightly outperforms random guessing is considered a weak classifier in AdaBoost.

The algorithm works by iteratively training weak classifiers on various subsets of the training data. At each iteration, the algorithm assigns a weight to each sample in the training data. The weight reflects the difficulty of classifying the sample correctly. The algorithm then trains a weak classifier on the weighted data and evaluates its performance. The weights are then updated based on the performance of the weak classifier, with misclassified samples being assigned higher weights. This process is repeated until the desired number of weak classifiers is reached.

To make a final prediction, AdaBoost combines the predictions of all the weak classifiers, weighted by their individual accuracy. The final prediction is the class label that receives the most votes.

AdaBoost has the benefit of being able to attain high accuracy with a minimal number of poor classifiers. It can also handle imbalanced datasets by weighting the samples or classes differently. Additionally, AdaBoost is less prone to overfitting than other algorithms and can handle both numerical and categorical data. However, AdaBoost can be sensitive to noisy data and outliers, which can affect the performance of weak classifiers. It can also be computationally intensive, particularly when dealing with large datasets.

IV. RESULT

Bagging (Bootstrap Aggregating) Classifier: Belonging to the family of ensemble learning algorithms and is a well-known machine learning method. It creates a powerful classifier by combining several base classifiers. Using bootstrapped samples, the base classifiers in Bagging are trained separately on various subsets of the training data.

The algorithm works by creating several bootstrap samples of the training data,which are obtained by randomly sampling the training data with replacement. Each bootstrap sample is used to train a base classifier independently. Predictions of base classifiers are then combined to make the final prediction. In Bagging, the final prediction is often made by majority voting, where the class that receives the most votes is selected as the final prediction.

Reduced variance of the base classifiers can increase the model's overall accuracy, which is one of the benefits of bagging. It can also handle imbalanced datasets by weighting the samples or classes differently. Additionally, Bagging is less prone to overfitting than other algorithms and can handle both numerical and categorical data. But when working with big datasets or lots of base classifiers, bagging can be computationally expensive. Furthermore, it might not function effectively on datasets with highly linked attributes.

Random Forest Classifier: Belonging to the family of ensemble learning algorithms and is a well-known machine learning method. It creates a powerful classifier by combining various decision trees. Using bootstrapped samples and a randomly selected portion of the features, the decision trees in Random Forest are trained independently on various subsets of the training data.

The training data are randomly sampled using replacement to provide a number of bootstrap samples, which are then used as input into the algorithm. An independent decision tree is trained with each bootstrap sample. Only a randomly chosen subset of the features is taken into account at each branch in the tree, though. Overfitting and the correlation between the decision trees are both decreased by this procedure.

Random Forest weights each decision tree's prediction based on its own accuracy to produce a final prediction. The class label that garners the most support is the chosen forecast.

Random Forest has the advantage of being able to handle missing values as well as categorical and numerical data. By giving different weights to the samples or classes, it may also manage unbalanced datasets. Random Forest can also achieve excellent accuracy even with a limited number of decision trees and is less prone to overfitting than other algorithms. But when dealing with large datasets or a lot of decision trees, Random Forest can be computationally demanding. Furthermore, it might not function well on datasets with highly correlated features.

Decision Tree Classifier: Is a well-known machine learning technique that creates a tree-like model of decisions and potential outcomes. The dataset's features are represented by internal nodes in the model's tree-like structure, decisions based on those features are represented by branches, and outcomes or classes are represented by leaves.

Gini Index and Entropy are two often employed metrics to determine the appropriate feature to use in a Decision Tree to divide the data.

Gini Index is a measure of impurity or randomness that calculates the probability of a randomly chosen sample being incorrectly classified according to the distribution of the classes. The Gini Index of a dataset is calculated as follows:

Gini = 1 - ∑(p_i) ^2

where p is the proportion of samples belonging to a particular class i.

Entropy is another measure of impurity that calculates the amount of information or uncertainty in a dataset. The entropy of a dataset is calculated as follows:

Entropy = - ∑(p_i * log(p_i))

where p is the percentage of samples that fall under a certain class i.

To build the Decision Tree Classifier, the algorithm uses the selected measure (Gini Index or Entropy) to calculate the impurity of the dataset and then recursively selects the most advantageous characteristic to divide the data according to information gain. Information gain measures the reduction in impurity achieved by splitting the data based on a particular feature.

The technique keeps dividing the data until all samples belong to the same class or until a stopping requirement, such as a maximum depth of the tree or a minimal number of samples needed to split a node, is met. One of the advantages of Decision Tree Classifier is its ability to handle both numerical and categorical data, as well as missing values. It can also handle imbalanced datasets by weighting the samples or classes differently. Additionally, Decision Tree Classifier can provide interpretability, as the resulting tree can be visualized and understood. However, Decision Tree Classifier can be sensitive to noisy data and overfitting, particularly when the tree is too deep or too complex. It can also be computationally intensive, particularly when dealing with large datasets.

TABLE 1. PERFORMANCE COMPARISON OF DIFFERENT CLASSIFIER MODELS

	Model	Accuracy	Precision	Recall	F1-score
1	Random Forest	0.52	0.62	0.34	0.44
2	Naive Bayes	0.58	0.65	0.58	0.62
3	K_Nearest Classifier	0.51	0.62	0.44	0.51
4	Logistic Regression	0.58	0.70	0,56	0.62

5	Gradient Boosting	0.51	0.65	0.40	0.49
6	Support Vector Machine	0.58	0.72	0.47	0.57
7	Decision Tree	0.47	0.50	0.43	0.40
8	Bagging Classifier	0.52	–	–	–
9	Ada Boosting	0.46	0.43	0.58	0.49

V. CONCLUSION

In this study, we looked into various machine learning methods for sentiment analysis of tweets, both positive and negative. The goal variable of sentiment was included with the dataset of 1000 tweets with 4 features. To examine the effectiveness of various classifiers, we used data preparation, feature selection, and model evaluation procedures. Our research shows that Naïve Bayes and SVM are most accurate algorithms for this task with accuracy rates of 58% each respectively. Our findings can be used in numerous applications in social media monitoring, brand management, customer service, political analysis, and more. By analysing the sentiment of tweets, businesses can gain insights into customer behavior, improve customer engagement, and enhance brand reputation. In order to increase the forecast accuracy and dependability, we propose that future study concentrate on enhancing the data quality, adding more features, and utilizing more sophisticated machine learning techniques.

REFERENCES

[1] Pak, Alexander, and Patrick Paroubek. "Twitter as a corpus for sentiment analysis and opinion mining." LREc. Vol. 10. No. 2010. 2010.

[2] Go, Alec, Lei Huang, and Richa Bhayani. "Twitter sentiment analysis." Entropy 17 (2009): 252.

[3] Wang, Hao, et al. "A system for real-time twitter sentiment analysis of 2012 us presidential election cycle." Proceedings of the ACL 2012 system demonstrations. 2012.

[4]. Sanders, Abraham C., et al. "Unmasking the conversation on masks: Natural language processing for topical sentiment analysis of COVID-19 Twitter discourse." AMIA Summits on Translational Science Proceedings 2021 (2021): 555.

[5] Basiri, Mohammad Ehsan, et al. "A novel fusion-based deep learning model for sentiment analysis of COVID-19 tweets." Knowledge-Based Systems 228 (2021): 107242.

[6] Aiswarya, M. K. "Sentiment Analysis of Twitter using Machine Learning." Journal of Research Proceedings 1.2 (2021): 216- 225.

[7] Giachanou, Anastasia, and Fabio Crestani. "Like it or not: A survey of twitter sentiment analysis methods." ACM Computing Surveys (CSUR) 49.2 (2016)

[8] El Rahman, Sahar A., Feddah Alhumaidi AlOtaibi, and Wejdan Abdullah AlShehri. "Sentiment analysis of twitter data." 2019 international conference on computer and information sciences (ICCIS). IEEE, 2019

[9] Neethu, M. S., and R. Rajasree. "Sentiment analysis in twitter using machine learning techniques." 2013 fourth international conference on computing, communications and networking technologies (ICCCNT). IEEE, 2013.

[10] Sebastian, T. (2012). Sentiment Analysis for Twitter (Doctoral diss ertation).

[11] Kodipalli, A., Guha, S., Dasar, S., & Ismail, T. (2022). An inception-ResNet deep learning approach to classify tumours in the ovary as benign and malignant. Expert Systems, e13215.

[12] Ruchitha, P. J., Richitha, Y. S., Kodipalli, A., & Martis, R. J. (2021, December). Segmentation of Ovarian Cancer using Active Contour and Random Walker Algorithm. In 2021 5th International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques (ICEECCOT) (pp. 238-241). IEEE.

[13] Kodipalli, A., Devi, S., Dasar, S., & Ismail, T. (2022). Segmentation and classification of ovarian cancer based on conditional adversarial image to image translation approach. Expert Systems, e13193.

[14] Ruchitha, P. J., Sai, R. Y., Kodipalli, A., Martis, R. J., Dasar, S., & Ismail, T. (2022, October). Comparative analysis of active contour random walker and watershed algorithms in segmentation of ovarian cancer. In 2022 International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER) (pp. 234-238). IEEE.

[15] Gururaj, V., Ramesh, S. V., Satheesh, S., Kodipalli, A., & Thimmaraju, K. (2022). Analysis of deep learning frameworks for object detection in motion. International Journal of Knowledge-based and Intelligent Engineering Systems, 26(1), 7-16.

[16] Guha, S., Kodipalli, A., & Rao, T. (2022). Computational Deep Learning Models for Detection of COVID-19 Using Chest X-Ray Images. In Emerging Research in Computing, Information, Communication and Applications: Proceedings of ERCICA 2022 (pp. 291-306). Singapore: Springer Nature Singapore.

[17] Rachana, P. J., Kodipalli, A., & Rao, T. (2022). Comparison Between ResNet 16 and Inception V4 Network for COVID-19 Prediction. In Emerging Research in Computing, Information, Communication and Applications: Proceedings of ERCICA 2022 (pp. 283-290). Singapore: Springer Nature Singapore.

[18] Zacharia, S., & Kodipalli, A. (2022). Covid Vaccine Adverse Side-Effects Prediction with Sequence-to-Sequence Model. In Emerging Research in Computing, Information, Communication and Applications: Proceedings of ERCICA 2022 (pp. 275-281). Singapore: Springer Nature Singapore.

[19] Kodipalli, A., Guha, S., Dasar, S., & Ismail, T. (2022). An inception-ResNet deep learning approach to classify tumours in the ovary as benign and malignant. Expert Systems, e13215.

[20] Kodipalli, A., Fernandes, S. L., Dasar, S. K., & Ismail, T. (2023). Computational Framework of Inverted Fuzzy C-Means and Quantum Convolutional Neural Network Towards Accurate Detection of Ovarian Tumors. International Journal of E-Health and Medical Communications (IJEHMC), 14(1), 1-16.

[21] Kodipalli, A. (2018). Cognitive architecture to analyze the effect of intrinsic motivation with metacognition over extrinsic motivation on swarm agents. International Journal of Electrical and Computer Engineering, 8(5), 3984.

[22] Kodipalli, A., & Devi, S. (2021). Prediction of PCOS and mental health using fuzzy inference and SVM. Frontiers in Public Health, 1804.

[23] Kodipalli, A., & Devi, S. Analysis of fuzzy based intelligent health care application system for the diagnosis of mental health in women with ovarian cancer using computational models. Intelligent Decision Technologies, (Preprint), 1-12.