An Intellectual Decision System for Classification of Mental Health Illness on Social Media using Computational Intelligence Approach

M. Sujithra[1], J. Rathika[2], P. Velvadivu[3], M. Marimuthu[4], Abhinaya.V[5], Reshma. R[6]

[1,2,3,4]Assistant Professor, Department of Computing (Data Science), Coimbatore Institute of Technology, India.

[5, 6]Student, Department of Computing (Data Science), Coimbatore Institute of Technology, India.

E-mail: [1]sujithra@cit.edu.in, [2]jradhika@cit.edu.in, [3]velvadivu@cit.edu.in, [4]mmarimuthu@cit.edu.in, [5]vabhinaya2002@gmail.com, [6]reshrajendiran@gmail.com

Abstract

Despite the advantages, the significant increase in the use of social media has also reflected in causing various health consequences. Social media provides a platform for people to directly express their emotions about the products they buy and also make suggestions. As a result of this, the social media users are facing more decision-making challenges. The emotions obtained from social media are based on polarity ranking, but this measurement of emotions is insufficient in real-time. A novel model should be designed in such a way that it correctly categorizes the human emotions via social media. This research work has attempted to predict the human emotions based on their social media posts, comments, and so on. Here, the machine learning algorithms are used to intelligently classify the human emotions and provide a better decision-making model.

Keywords: Social media analysis, decision-making, computational intelligence, machine learning.

1. Introduction

Nowadays, humans are more attracted to the digital technologies. Social media has a significant impact on a person's mental health and happiness. When people are digitally connected, they may experience stress, anxiety, depression, and sadness. On the other hand, this could jeopardize the mental and emotional health of social media users. Various social

DOI: https://doi.org/10.36548/jucct.2023.1.002
Received: 28.12.2022, received in revised form: 28.1.2023, accepted: 11.2.2023, published: 24.2.2023

media platforms are available for finding and connecting with people, including YouTube, Facebook, Twitter, and Instagram. However, there are both advantages and disadvantages in using these connections. Some may become depressed as a result of some of the events, while others may relish some of the events. It has recently become impossible for humans to categorize the emotions expressed in social media. By obtaining the data from social media app, a model can be created to classify the users' mental health. To accomplish this, the proposed research work intends to use the supervised machine learning techniques. Here, supervised machine learning algorithms such as Logistic Regression, K Nearest Neighbors Classifier, Decision Tree Classifier, Random Forest Classifier, XGBoost Classifier are used to classify mental illnesses based on social media posts, as well as computational algorithms such as genetic algorithms.

2. Related Work

Iqra Ahmer, Muhammed Arif, et al [1] presents a comprehensive review based on mental health illness detection based on social media text using deep learning and transfer learning. This study mainly concentrates on applying deep learning, machine learning and transfer learning methods to solve multi class mental illness detection challenge. The best result has been obtained by using the pre-trained RoBETRa transfer learning with an accuracy of 0.83 and F1-score=0.83. Syed Nasrullah, Asadullah Jalali [2] detected mental illnesses through the social network by using ensemble deep learning model. This study discusses about the mental health illness of people through different social media platforms such as twitter and reddit. Further, different types of mental illnesses are classified by using a deep learning model. Deepali Joshi, Dr. Manasi Patwardhan [3] have analyzed the mental health illnesses that occur to the social media users by using an unsupervised approach. In this study, the unsupervised learning algorithms has been applied on the data by signaling behavior change for performing psychological analysis and also to identify the probability of users showing a risk behavior.

George Gkotsis,Anika Oellrich [4] proposed a comprehensive study on characterization of mental health conditions of social media users by deploying advanced deep learning technologies. In this study, the post from Reddit has been analyzed and classifiers are developed to recognize and classify the social media post related to metal illness by using a neural network model with an accuracy of 71.37%.

Mohsin Kamal, Subhan Tariq [5] predicted the metal health illness of different social media users and proposed a novel XGBoost model to perform for accurate classification of obtained social media data based on four mental disorders like Schizorphrenia, Autism, OCD, PSTD. The result of the proposed methodology is then compared with Naive Bayes (NB) and Support Vector Machine (SVM). The 68% accuracy has been achieved by indicating the efficacy of the proposed model.

3. Proposed Work

Nowadays, social media has become the most common communication platform where people can express their feelings, emotions, and comments to others. However, posting comments for images and videos is not always done politely. This may sometimes affect the mental health of the users. As a result, the majority of people on social media experience some mental illness. Machine learning and computational intelligence technology are recently proposed to classify those with mental illnesses on social media.

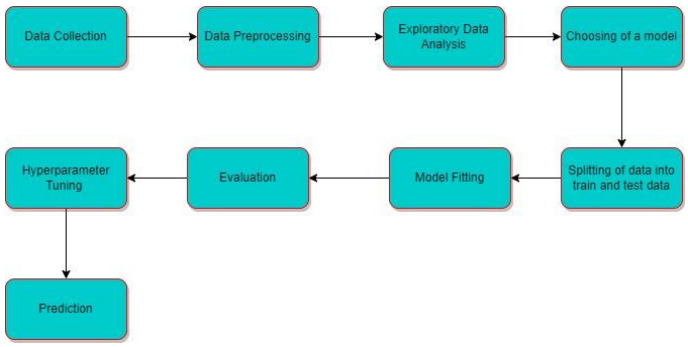

Figure 1. Workflow of the proposed methodology

The above figure 1.1 shows the workflow on how the data has been collected from different sources, data pre-processing, visual representation and then training and categorization of the final processed data.

The dataset is taken from Kaggle website. The attributes that are included in the dataset are subreddit, post_id, text, label, confidence, social_timestamps, syntax_ari, and lexical columns. The dataset consists of more than 10 features with more than 20,000 records.

Subreddit - a particular topic that people write about followers by their reddit name.

Post_id - the id of the post which the user has posted in reddit

Text - the text of comments which the other reddit users has posted for the other users' posts.

Label - 1 - (for stress), 0 - (for no stress)

Confidence - Confidence value for undergoing stress

Link: https://www.kaggle.com/datasets/ruchi798/stress-analysis-in-social-media

3.1 Sample Dataset:

Table 1: Reddit Dataset

	subreddit	post_id	sentence_	text	id	label	confidenc	social_timestamp	social_karma
2	ptsd	8601tu	(15, 20)	He said he	33181	1	0.8	1521614353	
3	assistance	8lbrx9	(0, 5)	Hey there	2606	0	1	1527009817	
4	ptsd	9ch1zh	(15, 20)	My mom t	38816	1	0.8	1535935605	
5	relationsh	7rorpp	[5, 10]	until i me	239	1	0.6	1516429555	
6	survivorsc	9p2gbc	[0, 5]	October is	1421	1	0.8	1539809005	
7	relationsh	7tx7et	(30, 35)	I think he	17554	1	1	1517274027	
8	domestic\	7iphly	[25, 30]	It was a bi	165	0	0.8	1512854409	
9	anxiety	5m3k80	(5, 10)	It clearec	33053	1	0.8	1483582174	
10	relationsh	7nhy1v	(50, 55)	I actually	7581	1	0.6	1514843984	1
11	assistance	61eiq6	[15, 20]	I just feel	1517	1	1	1490428087	
12	anxiety	8sxdmy	[0, 5]	Next wee	1156	1	0.571429	1529630854	
13	anxiety	87l3pf	[20, 25]	Everythin	1929	1	1	1522177346	
14	anxiety	910inu	(0, 5)	Iâ€™ve be	48347	1	1	1532289383	
15	homeless	8l9pp2	(1, 6)	No place i	4687	1	1	1526993242	
16	anxiety	9wypos	(4, 9)	These pas	15182	1	1	1542188956	
17	relationsh	7p6dgo	(20, 25)	I asked hir	48068	1	0.833333	1515495261	4
18	anxiety	8mqmx3	[35, 40]	The sensa	901	1	0.8	1527519277	
19	domestic\	815k8h	[2, 7]	It did hurt	1055	1	1	1519914067	
20	assistance	7rfjom	(10, 15)	This is hel	8594	0	1	1516332924	
21	relationsh	7pg3a7	[150, 155]	We had pl	2223	0	0.6	1515596406	
22	domestic\	79tofd	[5, 10]	he still ho	1985	1	0.8	1509426703	
23	survivorsc	9xkqh0	(10, 15)	I was goin	42454	0	0.8	1542360878	
24	ptsd	9jhgob	[0, 5]	Through w	1220	1	0.6	1538087875	
25	ptsd	91oz3r	[0, 5]	With a sw	138	0	0.571429	1532495820	

Table 1 represents the sample dataset that has been collected from the reddit data.

3.2 Data Acquisition:

Data acquisition is the process of collecting real-time information. Data is a set of raw facts, and data acquisition is the method of collecting data from various sources. The dataset in this proposed model is sourced from Kaggle. Once the data has been acquired, it is moved to the preprocessing stage, where it is visually presented to determine how the data is distributed or discover insights in the data through exploratory data analysis. Following that, the data is provided for model construction through classification algorithms.

4. Analysis

4.1 Exploratory Data Analysis

Exploratory data analysis (EDA) is used to analyze and investigate data sets and bring out useful insights from those data so that it helps people to easily understand how the data is getting distributed as well as useful insights from it rather than seeing it as a numerical value.

The boxplot for the attributes like label and confidence has been visually presented to check how the data distribution is between these two features.

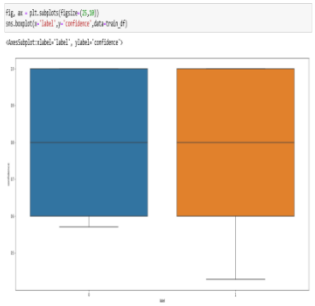

Figure 2. BoxPlot for label and confidence feature

The above figure 2 represents the box plot for the two label “Stress” and “Not Stress”. The X-axis represents the label value and Y-axis represents the confidence value. Since the text attribute has been included in this proposed model, word cloud visualization techniques are used here to visually represent the frequently appearing words in the particular text attribute. As a first step, the text with stress as a label was included. The word cloud was then used to present the information visually.

```
wordcloud = WordCloud(width=1000, height=500, stopwords=stopwords).generate(label_1_para
plt.figure(figsize=(15,15))
plt.imshow(wordcloud)
plt.axis('off')
plt.show()
```

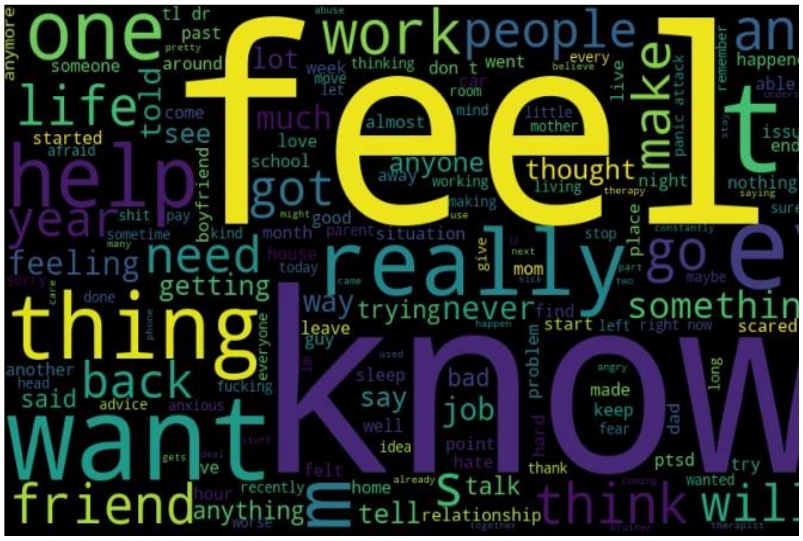

Figure 3. Word Cloud for the label stress

Figure 3 represents the word cloud represents the frequency of the words that are occurred in the text which has stress as a labels

```
wordcloud1 = WordCloud(width=1000, height=500, stopwords=stopwords).generate(la
plt.figure(figsize=(15,15))
plt.imshow(wordcloud1)
plt.axis('off')
plt.show()
```

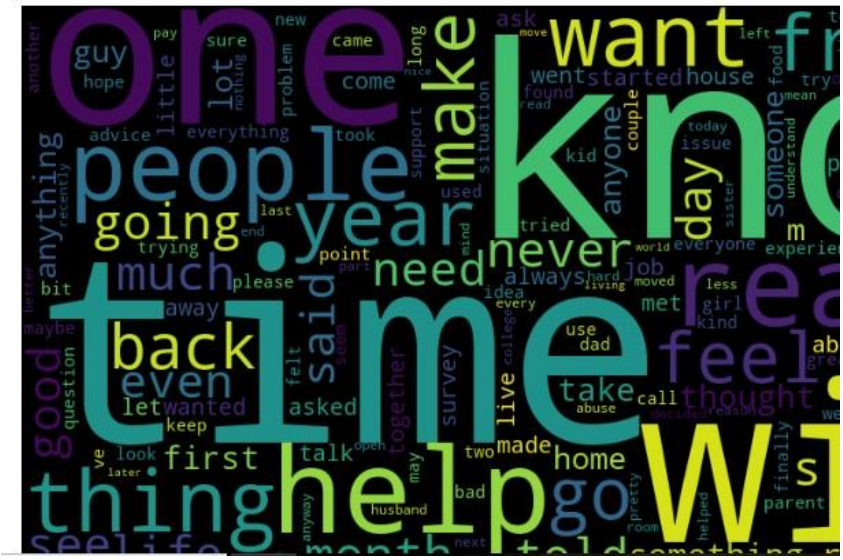

Figure 4. Word Cloud for label Not-Stress

Figure 4 represents the word cloud represents the frequency of words that occurs in the text feature which has a label as a not stress.

4.2 Preprocessing on Reddit Dataset:

Preprocessing is a technique that is used to remove the unnecessary noise and outliers from the data that needs to be analyzed. In this proposed model, the dataset is initially checked for null values. If null values are found and if the significance of the row values is low, the rows are dropped using dropna() function.

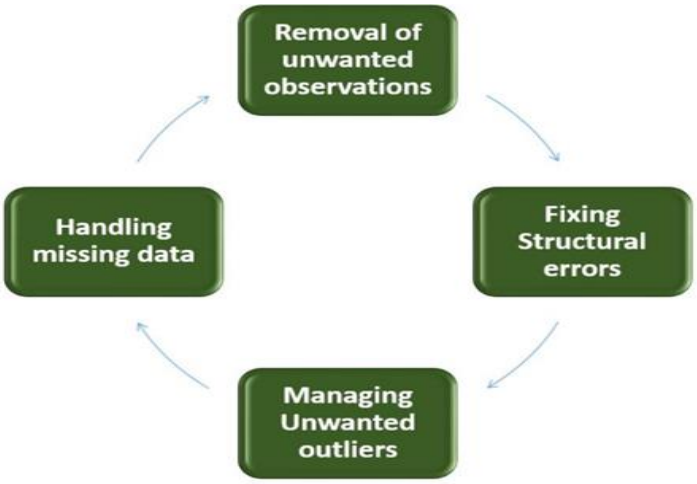

Figure 5. Pictorial Representation of Preprocessing Techniques

Checking for Null Values in the Dataset

```
train_df.isna().sum()
```

The function mentioned above represents the checking for the null values in the dataset using the function dropna().

Feature Scaling of Attributes

Next, as a preprocessing step a stopwords techniques has been used where the unimportant or the word that does not add much meaning to the sentence has been removed by making use of stop words library. As a next preprocessing step, the feature scaling technique has been used where the MinMaxScaling() and Normalisation() technique has been implemented. Normalization technique has been used to bring out the attribute values to lie into a particular range. In this dataset the attributes like sentiment, confidence,lex_dal_avg_pleasantness etc.

```
#Feature Scaling
minmax=MinMaxScaler()
stdscaler=StandardScaler()
```

The aforementioned are the feature scaling function, where the feature scaling for the features in the dataset is done using MinMaxScaler() function.

Label Encoding for Categorical data

The next preprocessing step includes label encoding where the categorical column is converted in to numerical column. Here the attribute subreddit is converted into numerical column by making use of LabelEncoding() technique.

```
#Label Encoding
label_enc=LabelEncoder()
train_df[‘subreddit’]=label_enc.fit_transform(train_df[‘s
ubreddit’])
```

The above representation shows the label encoding done for the categorical data where the categorical values get converted into numerical values.

Feature Selection

Feature Selection technique has been implemented where the highly correlated features has been dropped by using variance threshold value where the attributes that has the variance value below the threshold value is considered unimportant and those variables has been dropped.

Almost all of the features in the dataset have no impact on model fitting. Even if all of the features are included, the accuracy may suffer or the result may not be as expected. To overcome this, a feature selection technique for performing effective model fitting must be implemented. As a result, the techniques are used to include variance threshold and mutual data classifier, where significant features are selected and high correlated features are excluded. The variables are also mutually selected by using mutual_info_classif() and SelectKBest() function, where the mutual features in the dataset are selected.

5. Implementation

For the reddit dataset the classification model has been encountered to classify the mental health illness of a people on a social media. This classification model includes Logistic Regression, K Nearest Neighbor Classifier, Decision Tree Classifier, Random Forest Classifier and XGBoost Classifier. Once the model has been fitted in order to find the accuracy of each model, precision, recall and F1 score have been calculated for the respective algorithms in order to find the best fit model.

5.1 Importing Necessary Libraries

```
from sklearn.linear_model import Logistic Regression
from sklearn.neighbors import KNeighborsClassifier
from sklear.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from xgboost import XGBClassifier
from sklearn.metrics importaccuracy_score,classification_report,confusion_matrix
```

In the above Fig 5.1 code, the necessary libraries have been which are needed to fit the model and for further performance, the model has been fitted

5.2 Allocating Dependent and Independent Variable

```
X=train_df.drop('label',axis=1)
Y=train_df ['label']
```

Here, the dependent and independent variable are allocated for the respective X and Y variable. Once the loading and preprocessing are done, the next step is to split the data. In the above code the dataset is split into two parts training data and then testing data and it is named X_train and Y_train and then is to be fitted into classification models and the output is stored for different models separately.

5.3 Splitting the data into train and test data

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.25)
X_train = np.asarray(X_train)
X_test = np.asarray(X_test)
Y_train=np.asarray(Y_train)
Y_test=np.asarray(Y_test)
```

The aforementioned code represents the splitting of data into training and test data.

5.4 Fitting of Logistic Regression

```
X_chi=X[chi_feature]
best_model=(X_chi , y ,minmax, Logistic Regression)
```

The code mentioned above is implemented for the Logistic Regression (LR) model. Here, the training and the testing data are fitted in to the model and the result is stored in the variable called predictions.

5.5 Fitting of K-Neighbors Classifier

```
X_chi=X[chi_feature]
best_model=(X_chi , y ,minmax, KNeighborsClassifier)
```

The above Fig 5.5 represents the model which is fitted here is K-Nearest Neighbors Classifier, where the training and the testing data is fitted into this model. The result is stored under the variable predictions.

5.6 Fitting of Decision Tree Classifier

```
X_chi=X[chi_feature]
best_model=(X_chi , y ,minmax, DecisionTreeClassifier)
```

The above code represents is a decision tree classifier, where the training and testing data is fitted in to the model and the predicted output gets stored under the variable prediction.

5.7 Fitting of XGBoost Classifier

```
X_chi=X[chi_feature]
best_model=(X_chi , y ,minmax, XGBoostClassifier)
```

The above code represents the fitting of XGBoost Classifier into the training data.

5.8 Fitting of Fitness Function

```
def fitness_func(solution,solution_idx):
output=numpy.sum(solution*list1)
fitness=1.0/numpy.abs(output-desired_output)
return fitness
```

Here, the computational intelligence algorithm called evolutionary algorithm has been fitted by making use of fitness function which takes the candidate solution to the problem as input and produces as output how "fit" our how "good: the solution is with respect to the problem that has been taken into consideration.

6. Model Evaluation and Validation

A model evaluation is a process of checking the performance of the model. For the training model, how the model is working or performing for the test data. The evaluation of the model is done and is evaluated based on its performance metrics like precision, recall, F1-score etc. Based on it the best model can be chosen for the proposed model so that the performance can be enhanced.

6.1 Evaluation Metrics for Classification Model:

Table 2: Performance Metrics for Logistic Regression

	Precision	Recall	F1-Score	Support
0	0.74	0.72	0.73	340
1	0.75	0.77	0.76	370

The above table 2 shows the result is the performance metrics for the Logistic Regression Classifier. The Accuracy that is obtained from the above model is 74.78%.

Table 3: Performance Metrics for K Nearest Neighbors Classifier

	Precision	Recall	F1-Score	Support
0	0.76	0.62	0.68	356
1	0.68	0.81	0.74	354

The above table 3 is the result of the performance metrics of the K Nearest Neighbors classification model and the accuracy obtained is 71.12%

Table 4: Performance Metrics for Decision Tree Classifier

	Precision	Recall	F1-Score	Support
0	0.64	0.66	0.65	329
1	0.70	0.68	0.69	381

The above table 4 shows the results thus obtained is the performance metrics for Decision Tree Classifier where the accuracy thus obtained by using this model is 66.90%

Table 5: Performance Metrics for Random Forest Classifier

	Precision	Recall	F1-Score	Support
0	0.75	0.72	0.74	341
1	0.75	0.78	0.77	369

The above table 5 is the result thus obtained in the performance metrics for the Random Forest Classifier where the accuracy thus obtained is 75.12%.

Table 6: Performance Metrics for XGBoost Classifier

	Precision	Recall	F1-Score	Support
0	0.78	0.68	0.72	355
1	0.71	0.81	0.76	355

From table 6, it is evident that the accuracy obtained for the XGBoost Classifier model is 74.22%.

7. Interpretation

As mentioned above, the increase in social media led people to show their emotions, feelings on the post of the others or the products which they buy. Thus, by making use of appropriate machine learning model, the classification on the people's feelings can be classified and assist the person, who sell their products to know how people feel about their products or the people, who post their images to know the comments of others etc. Here, different classification methods such as Logistic Regression, K Nearest Neighbors Classifier, Decision Tree Classifier, Random Forest Classifier and XGBoost Classifier, Evolutionary Algorithm are used. By fitting the training data into this algorithm, the model has been implemented and then the prediction and validation of data is done by using the test and validation data. Thus, the results are then derived by analyzing the accuracy, precision and recall of each model as follows:

Logistic Regression has resulted in an accuracy of 0.74 and with the precision for positive class is 0.76 and for negative class is 0.73. The accuracy of K-Nearest Neighbors classifier is 0.71 and the precision for positive class is 0.74 and for negative class is 0.68. The accuracy of Decision Tree Classifier is 0.66 with the precision of positive class is 0.69 and of negative class is 0.65. The accuracy for the Random Forest (RF) classifier and XGBoost Classifiers is 0.74 and 0.74 respectively. For the Evolutionary Algorithm, the predicted output based on the best solution is 1.04.

Thus, by using the above algorithms, the best model which classifies the data accurately is random forest classifier, which has the highest accuracy of 75.12% compared to

other models. As a result, the Random Forest (RF) classifier model can be chosen to classify the mental health of the people based on their comments and replies on the particular post on social media.

From this we can infer that, for the classification model, the Random Forest (RF) classifier has been selected as the best model with an accuracy of 75.12%. We can also infer that the precision and recall has also been chosen as the performance metrics for calculating the performance for the particular classification model.

8. Conclusion

Social media is a great source of communication and emotions from their post and comments can be predicted by using machine learning algorithms. Here, in the proposed model, the mental health in social media has been focused wherein the classification of datasets has been included and then the corresponding machine model has also been fitted. For classification, the machine learning model that has been used here is Naive Bayes (NB) classifier, Random Forest (RF) classifier, Decision Tree (DT) classifier and XGBoost classifier. For this, the model evaluation and performance metrics has been established. Among those classification models, the Random Forest (RF) model has been considered as the best fit model because of its higher accuracy level with 75.12% which is compared to all other models.

References

[1] Abusaa, M., Diederich, J., Al-Ajmi, A., et al.: Machine learning, text classification and mental health. HIC 2004: Proceedings p. 102 (2004).

[2] Ameer, Iqra, Noman Ashraf, Grigori Sidorov, and Helena Gómez Adorno. "Multi-label emotion classification using content-based features in Twitter." Computación y Sistemas 24, no. 3 (2020): 1159-1164.

[3] Benton, Adrian, Margaret Mitchell, and Dirk Hovy. "Multi-task learning for mental health using social media text." arXiv preprint arXiv:1712.03538 (2017).

[4] Amjad, Maaz, Grigori Sidorov, and Alisa Zhila. "Data augmentation using machine translation for fake news detection in the Urdu language." In Proceedings of the 12th language resources and evaluation conference, pp. 2537-2542. 2020.

[5] Mani Barathi SP S , Rekha V S , Dr.M. Sujithra , Dr.K. Rajarajeshwari,“Depression and Anxiety Detection with Machine Learning Techniques using Data from Wearable Sensors “,YMER journal, VOLUME 21 : ISSUE 9 (Sep) – 2022.

[6] Sujithra, M., P. Velvadivu, J. Rathika, R. Priyadharshini, and P. Preethi. "A Study On Psychological Stress Of Working Women In Educational Institution Using Machine Learning." In 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-7. IEEE, 2022.