ISSN 2278-3091
Volume 9, No.4, July – August 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse95942020.pdf
https://doi.org/10.30534/ijatcse/2020/95942020

Intelligent Decision Support System for Disaster Managers using Machine Learning

A. Yovan Felix[1], T. Sasipraba[2], Chilamkurthi Veera Srikanth[3], Daggubati Sujay[4], Chaduvula Ravi Kiran[5]
[1]Research Scholar, School of Computing, Sathyabama Institute of Science and Technology, Chennai-600119
[2]Professor, School of Computing, Sathyabama Institute of Science and Technology, Chennai-600119
[3,4,5]UG Student, School of Computing, Sathyabama Institute of Science and Technology, Chennai-600119

ABSTRACT

Due to various natural disasters, the entire world is prone to disasters like storms, floods, tsunami etc. Flood is one of the major disasters that affect the economy as well as the inhabitants. In this paper, we addressed the Cuddalore district flood. Geographically, it is located in the coastal region and has four rivers in Tamil Nadu. Prior to the disaster, the meteorological department warns the possible flood-prone regions. The respective district collector along with the local bodies communicate with each other and take decisions, which is a time-consuming process. In this aspect, an essential tool is required in order to reduce this impact. Cuddalore Flood Management DSS (CFMDSS) is an intelligent DSS developed with the state of art machine learning algorithms with high accuracy and precision which makes this process much simpler. Data from various departments like meteorological and hydrological are collected to train the model. The live data is captured through API's and this DSS will be able to detect the disaster prior 5 days. The flood-prone areas are neatly mapped in the web-based GIS which can be extensively visualized and accessed free of cost. The system is tested against the real scenarios and proven to give accurate results.

Key words : Cuddalore, Flood, DSS, Machine Learning, Flood disaster management.

1. INTRODUCTION

Natural Disasters are adverse unforeseen events that are provoked by the natural process. These extreme environmental circumstances, not only affect the social and economic factors but also takes so many lives. In July 1931 China has seen a most deadly flood (Yangtze-Huai River) which took over 4 million lives, within 1953 and 2016, the Central Water Commission figured that floods held responsible for nearly 1 lakh deaths in India, and property loss of around 35 million. In future, climate change is expected to raise the frequency and severity of natural hazards, heading to serious outcomes and disturbances to the society and the environment [1]. Predicting these disasters in prior is a tough task. So such kind of situation, a Powerful, accurate and Intelligent Decision Support System(DSS) which is capable of detecting flood in prior [9] to a considerable amount of time is mandatory.

In this research, data is acquired from various departments like meteorological and hydrological departments within the years 2006 and 2016. Among the various parameters obtained, 11 important features are identified for the analysis. At this stage, preprocessing is performed in order to reduce imbalance nature. Now the data is split into 70-30 and SMOTE (Synthetic Minority Over-Sampling Technique), a preprocessing algorithm is used to balance the training data (70%). This preprocessed data is used for training using XGBoost algorithm and 30% of the data is used to validate it. The evaluation metrics used are F1 score, precision and recall. The Machine Learning model is generated and the live data from the API is taken and given to the model so that it will give the result whether flood or non-flood based on the training given to it.

The Machine Learning Model is a complex mathematical and condition based artefact that is created by a machine learning algorithm that takes input and gives output based on the training given to it.

2. LITERATURE SURVEY

ANFAS [5] is a Decision Support System developed as a joint project for the People's republic of China and the EU. The Web-based distributed architecture enables the flood managers to simulate what-if scenarios and estimate the potential impacts. Some of the main Components include Data Fusion, Data Assimilation, GIS Modelling for Visualization. DELFT-FEWS [4] is another DSS which provides state of the art Flood Forecasts and Warning system. It is an Open data handling platform designed for hydrological forecasting system and the data can be used for viewing, exporting, importing and manipulation. The Complexity of the model is handled by Open approach model protocols. DESYCO [8] is a GIS-based DSS for assessing multiple climate change impact on Coastal regions. The main idea is the RRA (Regional Risk Assessment) methodology

that ranks the potential targets and the areas at risk from climate change. MCDA (Multi-Criteria Decision Analysis) is used to identify the areas which were exposed to climate-related hazards. The GUI-based system provides users with a step-by-step application of DESYCO.

FIDSS (Flood Integrated DSS) is developed by Melbourne Water (MW). The main idea of the system is to integrate Telemetry systems, URBS Hydrological models and Flood maps. The system is capable to give precise flood warnings by BOM (Bureau of Meteorology) and Flood Zoom through Email alerts. FLIRE DSS [3] is a web-based DSS which serves as a multi-purpose system for both Forest fire control and Flood risk monitoring. What-if scenarios are implemented for areas that are prone to fires and Flood maps are used for Flood services. The system can be accessed through a PC or a Smartphone via a 3G or 4G network. FloodViz [7] is a Visual based DSS that uses Rainfall Thresholding Technique to identify the at-risk areas and provides this data through maps for extensive Visualization. It integrates Meteorological and Hydrological data to provide a description of the flood situation. The only parameter is the River Discharge which is not considered for Flood assessment. Flood Wise [6] is a Flash Flooding Emergency Tool that takes data from Telemetric gauges and displays the readings in a Web portal. The system integrates with Council GIS for visualization and automatic alerts are made via SMS or EMAIL to the on-call duty officers. Historic Playback is used to replay historic events. RAMFLOOD [2] is a DSS for Flood risk assessment and management. It combines Environmental data and Geophysical data with advanced computer simulations and Artificial Neural Networks (ANN).

3. ARCHITECTURE EXPLANATION

From the architecture diagram in the figure 1, the data is procured from the Indian Meteorological Department (IMD) and State ground and surface water resource data center i.e., Meteorological data and Hydrological data. Since the data is imbalanced i.e., non-flood events are very high compared to flood events, some feature engineering is done to reduce it. Even now data is imbalanced. The data is split into a 70-30 ratio for training and testing. SMOTE a pre-processing algorithm is applied to that 70% of data, have applied a pre-processing algorithm called SMOTE (Synthetic Minority Over-Sampling Technique) to balance it i.e., to make the flood and non-flood events in 1:1 ratio. This algorithm will generate synthetic samples based on existing minority data i.e., flood events. SMOTE comes under the category of over-sampling techniques. This balanced 70% data is utilized for training used XGBoost algorithm, it is a tree-based boosting ensemble Machine Learning algorithm which is good at dealing with imbalanced data. Hyper parameter tuning is done and the Machine Learning model is generated and tested against the remaining untouched imbalanced data i.e., 30%.

By using an API, the predicted live weather data of the next 5 days will be collected at regular intervals of time and fed to the model that is generated previously. This model will give the flood prediction in the form of binary 0(NO) or 1(YES) based upon the training that is given to it.

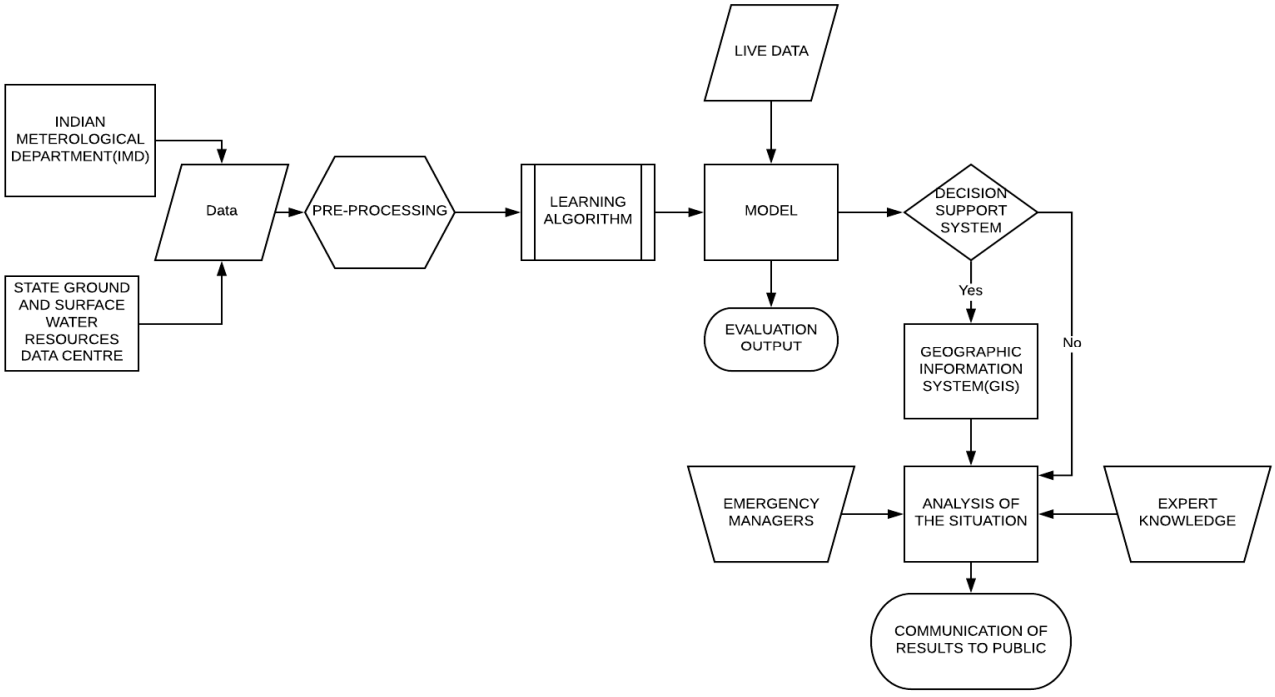

Figure 1: CFMDSS Architecture

4. METHODOLOGY

Requirements for Developing

In this research for developing the Decision support system, SMOTE (Synthetic Minority Over-Sampling Technique) a preprocessing algorithm is used to balance the data and Ensemble Machine learning algorithm for classification and python as a programming language (IPython) and packages used are NumPy, Pandas, Scikit-Learn, ImbLearner, Matplotlib.

This research mainly consists of two phases, in phase 1 development of a machine learning model from historical data. In phase 2 collecting live data from API and fed to that model and this will give output as a flood or a non-flood event.

Phase I

In this phase, machine learning is created and tested with the training data and tuned the parameters to get better results. First, the data is collected from the Indian Meteorological Department (IMD), State Ground and Surface Water Resources Datacenter, Public Work Department (PWD) which consists of features/ variables like

The data is from 2006-2016 and the readings are noted twice a day. Same units are followed throughout the data for the respective column and some of these variables are discrete and categorical. Out of all the features, 11 important dependent features are selected for our research (for detecting the flood). These selected features also consist of continuous and categorical data types.

To make this into a supervised learning problem the data is carefully and manually labelled the data by referring to District Disaster Management Authority, a new column is added and named as output where a flood event is marked as 1 and a non-flood as 0.

Analysis

The year parameter is removed because in every new year a new value would come and the algorithm would fail. While importing the data some of the categorical parameters are wrongly interpreted as discrete numeral values (like Month, Day) they are changed back to the categorical features as shown in figure 2. The Wind Direction has set of 16 different characters like N, S, W, E, NE, NNW etc., they are numerically encoded and converted [10] into categorical values so that the algorithm can learn from it

There are some missing values in the data, that was not filled with mean or median or mode because the Machine Learning algorithm is capable of filling the missing values on its own (provided the data type of the feature). The overall data looks like as in table 1. From the histogram, as shown in figure 3 the that highest value of rainfall is 176 but even then there is no flood and there are only 7 reading above 100 but none of them is flooded events.

Table 1: Data Description

	Month	Day	Min Temperature	Max Temperature	Temp Dry Bulb	Temp Wet Bulb	Relative Humidity	Av Wind Speed	Pan Evaporation	Rainfall
count	4048.000000	4048.000000	4048.000000	4048.000000	4048.000000	4048.000000	4048.000000	4048.000000	4048.000000	4048.000000
mean	9.005435	15.836957	25.436092	33.609684	29.596097	24.657856	67.686586	2.312794	2.673333	2.403542
std	2.155889	8.855321	3.197875	6.129185	3.657184	2.020229	16.881988	2.672785	1.600901	9.814202
min	6.000000	1.000000	0.300000	22.000000	20.000000	18.000000	22.000000	0.000000	0.000000	0.000000
25%	7.000000	8.000000	23.000000	31.000000	27.000000	24.000000	55.200001	0.000000	1.600000	0.000000
50%	9.000000	16.000000	25.000000	34.000000	29.000000	25.000000	67.800003	1.900000	2.400000	0.000000
75%	11.000000	23.250000	28.000000	36.000000	32.000000	26.000000	80.174999	3.900000	3.600000	0.000000
max	12.000000	31.000000	38.000000	336.000000	40.000000	38.000000	100.000000	14.770000	32.000000	176.000000

5. FEATURE ENGINEERING

The number of flood events totally are just 60 out of 8036. From the analysis, it is clear that the data is heavily imbalanced i.e., the ratio of non-flood events are very high when compared to flood events. SMOTE is used to balance the data but the results are not that fruitful because the data is heavily imbalanced. So, to solve this the months from the data where the flood has occurred supposed in the 11 years span has been selected. If the flood has occurred in February 2010 all the February month of the 11 years are considered, this would reduce the complexity and imbalance natured and yet gives good results by improving the evaluation parameters [11]. Even now the data is completely imbalanced with the percentage of Non-flood events: 99.9851778656 and percentage of 1's: 0.01482213438 but it is better than the previous step.

Month	categorical
Day	categorical
Min Temperature	float64
Max Temperature	int64
Temp Dry Bulb	float64
Temp Wet Bulb	int64
Relative Humidity	float64
Av Wind Speed	float64
Wind direction	categorical
Pan Evaporation	float64
Rainfall	float64
OutPut	categorical

Figure 2: Selected features and data types

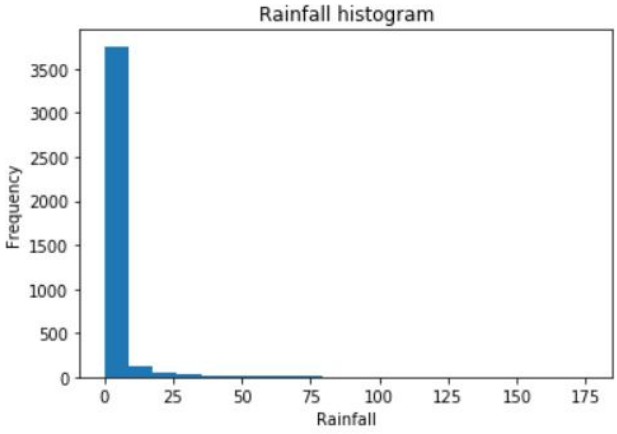

Figure 3: Histogram of rainfall

Before feature engineering the total number of events: 8036
Total flood events: 60
Percentage of flood events: Flood events/ (Flood events + Non-Flood events) 0.00746640119%
After Feature engineering the total number of events: 4048
Total flood events: 60
Percentage of flood events: 0.01482213438

Now that the data is prepared the model has to be built. The pipeline for building the model is:

5.1 Split the data into train and test part

70 to 30 ratio has been used for doing this, since, the data is imbalanced the stratified split i.e., the data is not simply divided into 70 to 30 but also the ratio of flood and non-flood events are respectively maintained proportionally in the individual split has been used.

5.2 Preprocessing

The preprocessing algorithm SMOTE is used to balance the data set. The figures 4 and 5 show the data in 2D-PCA (principal component analysis) plot for before and after applying the SMOTE. The yellow points represent the flood events and purple non-flood events. SMOTE is only applied to the training data (70%) but not to the test set because to prevent the data leakage. The SMOTE works in a way that it synthetically generates the samples of the minority class (flood events) and makes the data balanced i.e., the flood and non-flood events are in 1:1 ratio. The way it generates is from the available data it considers the flood events and generates new synthetic samples without removing the data (non-flood events, oversampling). It randomly considers two minority class points and draws a line between them, along that line at a random point it marks a new point, it does it until the data is balanced.

If SMOTE is applied to the entire data and then split it, then it won't be correct even the evaluation parameters are very high because the data is getting leaked. In the real world, the flood events are very less when compared to non-flood events so the 30% data which is used for testing is left as such to get the exact performance.

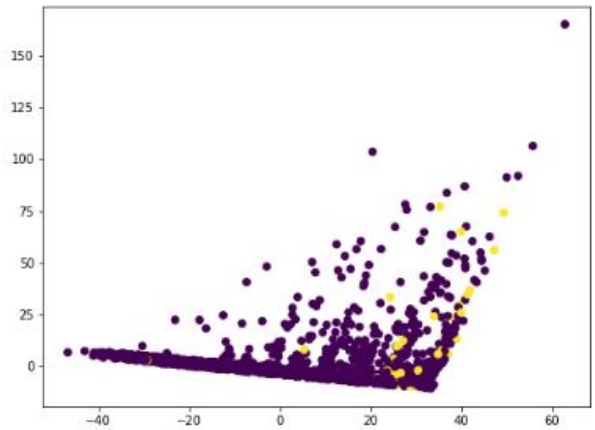

Figure 4: PCA Plot before Applying SMOTE

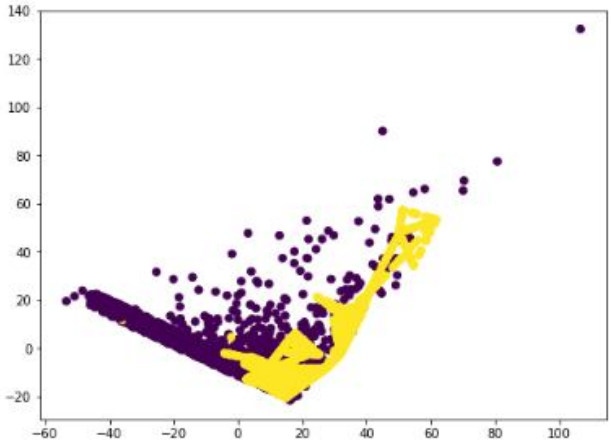

Figure 5: PCA Plot after Applying SMOTE

5.3 Generating Machine Learning Model

Before diving into ensemble algorithms the tried traditional algorithms like Decision Tree, K-Nearest Neighbor, Logistic Regression are tested but, the results are not up to the mark because they are not powerful enough to classify the minority class. Then we moved to ensembles these are nothing but the

combination of 100's or even 1000's of traditional algorithms. The Reason behind choosing these ensembles are these algorithms give good performance and are very powerful i.e., they are built in a way that they are forced to learn from the mistakes that are done in the previous steps. The XGBoost that is used is a tree-based boosting algorithm which is good at dealing with the imbalanced data compared to others. This model gives the probability of the occurrence of the flood. The custom threshold instead of standard 0.5 has been set and converted the output as binary.

5.4 Evaluation

As mentioned earlier the data is tested against the testing set which is left untouched by SMOTE and hyper parameters tuning is also done. The primary point to consider is accuracy is not being considered as our evaluation metric because it is an imbalanced problem the accuracy paradox occurs, the accuracy that is obtained for this system is above 99% because in test data very few flood events are present so even if the model predicts every instance as non-flood the accuracy will be very high this is called accuracy paradox. So for evaluation, the metrics like F score, precision and recall. Now the model which is tested against the real-world scenario is generated.

5.5 Phase II

The model has been tested with the historical data that is collected. But, the flood has to be predicted in the real-world scenario. For that, there is a need to capture the live data. So, the live data is captured using an API in python from a real-world weather forecasting engine which updates the data every 3 hours once (00:00, 03:00, 06:00, etc.,). Our system seamlessly calls the API at fixed time intervals. This data is given as input to the model. When the API is called, it will give the data of that time interval to the data of the next five days. The model predicts the flood for the given data based on the training given. So, this model predicts the flood 5 days prior to its occurrence.

6. RESULTS AND DISCUSSION

The model has been tested against test data (without applying SMOTE) and the following results are obtained.

TP= True Positive, FP= False Positive, FN=False Negative, TN=True Negative

Precision = TP/(TP + FP)

Recall = TP/(TP + FN)

F Score= 2 * Precision * Recall/(Precision + Recall)

Precision: 0.22727273

Recall: 0.55555556

F score: 0.32258065

These results show that the evaluation parameters are low, the values aren't changed any values (the data) that we have collected from the government recognized bodies. These evaluation parameters can be improved by adding more feature to the data like continuous water level readings of river, land use and land cover data. Yet a base has been established DSS in Cuddalore District by using Machine Learning Algorithms by utilizing the available data. This research can stand as a good foundation for further development

7. FUTURE WORK AND CONCLUSION

A decision support system for flood prediction which provides visualization of flood warning has been developed. The overcame the technical challenges like data imbalance by using feature selection, feature engineering and preprocessing algorithms and to deal with the real world imbalance data, ensemble boosting based machine learning techniques have been used and developed a model which is the heart of our DSS and tested against the real world data which is left untouched while splitting. This DSS will be helpful for the emergency flood managers so that the time will be reduced for the managers to take indispensable decisions. Our model is connected to a real-time weather forecasting engine through online which will be updated in a fixed time interval. So, the model can predict the flood 5 days prior to its occurrence. This helps in minimizing social and economic loss. At this stage, this system is not yet published on a public platform and it has not considered the river water level and land cover data. In future, the aim to revamp the system to improve the values of evaluation parameters and to connect the system to Web GIS to visualize the flood warning on a map.

Web GIS is a type of distributed information System, comprising at least a server and a client, where the server is a GIS server and the client is a web browser, desktop application, or mobile application. The main purpose of GIS is to give a visual representation of the output generated by the model. The predicted flood information will be given as input to GIS so that it can visualize the obtained output. The model will give the result by considering those live parameters. This output is given to Web GIS. Here, a web link will be given to the flood manager and if the flood is detected by the model, then the respective area will be marked, the flood managers can also include some messages from the map like the warning or some planning strategies.

REFERENCES

1. Caddis B, Kirby D, Minett A, Rasmussen P & Turnley M (2015) "A flood integrated decision support system for Melbourne". Floodplain Management Association National Conference, Brisbane, Australia.2015
2. Ernest Bladé Castellet, Manuel Gomez Valentin, Eugenio Oñate, Josep Dolz Ripolles, Georgina Teresa

Corestein Poupeau, Javier Piazzese(2006). "Decision Support System for Flood Risk Assessment and Management", 7thInternational Conference on Hydroinformatics – HIC 2006.

3. Giorgos Kochilakis; Dimitris Poursanidis; Nektarios Chrysoulakis; Vassiliki Varella; Vassiliki Kotroni; Giorgos Eftychidis; Kostas Lagouvardos; Chrysoula Papathanasiou; George Karavokyros; Maria Aivazoglou; Christos Makropoulos; Maria Mimikou (2016) "A web-based DSS for the management of floods and wildfires (FLIRE) in urban and periurban areas", Environmental Modelling and Software, Volume 86, Issue C, December 2016, Pages 111-115. https://doi.org/10.1016/j.envsoft.2016.09.016
4. M. Werner, J. Schellekens, P. Gijsbers, M. van Dijk, O. van den Akker, K. Heynert., 2012. "The Delft-FEWS flow forecasting system", 2012, DOI: 10.1016/j.envsoft.2012.07.010.
5. Poulicos Prastacos, William Castaings, Nathalie Courtois, Ladislav Hluchy, Patrick Houdry, Viet Tran (2005) "ANFAS: a decision support system for flood risk assessment ", e -environment: progress and challenge. vol. 11, pp. 61-80.
6. Robert McGlinn & Evan Caswel (2015), "Floodwise: A Flash Flooding Emergency Management Tool", Floodplain Management Association National Conference, Australia, 2015.
7. Sung, Er-Xuan & Tsai, Meng-Han & Kang, Shih-Chung. (2015). FloodViz: A Visual-Based Decision Support System for Flood Hazard Warning. DOI 10.22260/ISARC2015/0099.
8. SilviaTorresan, Andrea Critto, Jonathan Rizzi, Alex Zabeo, Elisa Furlan, Antonio Marcomini., 2015. "DESYCO: A decision support system for the regional risk assessment of climate change impacts in coastal zones". https://doi.org/10.1016/j.ocecoaman.2015.11.003.
9. A Yovan Felix, Sasipraba, T. (2016). "Incident mapping and EAS using decision support system", ARPN Journal of Engineering and Applied Sciences, vol. 11, NO. 15, August 2016, pp 9266-9269.
10. A. Y. Felix and T. Sasipraba, "Flood Detection Using Gradient Boost Machine Learning Approach," 2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE), Dubai, United Arab Emirates, 2019, pp. 779-783
11. Felix, A.Y., Sasipraba, T. Spatial and temporal analysis of flood hazard assessment of Cuddalore District, Tamil Nadu, India. Using geospatial techniques. J Ambient Intell Human Comput (2020). https://doi.org/10.1007/s12652-020-02415-y
12. Munya A. Arasi, Sangita babu, "Survey of Machine Learning Techniques in Medical Imaging", International Journal of Advanced Trends in Computer Science and Engineering, Volume 8, No.5, September - October 2019, pp. 2107-2116. https://doi.org/10.30534/ijatcse/2019/39852019
13. Thein Yu, Khin Thandar Nwet, "Myanmar News Sentiment Analyzer using Support Vector Machine Algorithm", International Journal of Advanced Trends in Computer Science and Engineering", 8(6), 2019, pp.3520-3525. https://doi.org/10.30534/ijatcse/2019/131862019