DIAGNOSIS OF ALZHEIMER DISEASE USING MACHINE LEARNING APPROACHES

B Venkanna
Assistant Professor, Department of CSE, Brilliant grammar school educational society group of institutions, Hyderabad, India
Email Id: venkanna559@gmail.com

***Abstract* - This article applies the machine learning paramount techniques for the early detection and effective diagnosis of severe Alzheimer's disease (AD). AD is neurodegenerative chronic disease and often essential to detect at the early stage. It is vital to diagnose the disease in initial stages for more effective and beneficial treatment. Machine learning is becoming the booming area and shows the remarkable achievement in the present, advance and crucial decision making. Medical diagnosis is one of the crucial areas that have paramount importance where various learning algorithms can be contributed for the improvement in disease diagnosis. Due to the evolution of computation technology, the generation of data is increased exponentially, especially in medical field. To cope up this problem, this paper elaborates and tests various approaches of machine learning for AD diagnosis. Using oasis longitudinal MRI data, this study train various models of machine learning to detect patients suffering from the AD. Along with the machine learning, this letter addressed the deep learning comprehensive overview. With various learning approaches such like binary classifier named logistic regression (LR), support vector machine (SVM), hierarchical decision tree (DT), ensemble random forest (RF), and boosting adaboost, experimental result is analyzed in terms of accuracy, recall, and AUC (Area Under Curve). The results obtained concluded that random forest and adaboost achieve higher accuracy, together with random forest also able to get higher recall and AUC. The minimum time to accomplish the classification task is taken by decision tree that is 68.788ms. This document result would be helpful to strengthen the idea and concept of applying the learning algorithms in disease detection at early stage.**

***Keywords:* logistic regression (LR), support vector machine (SVM), hierarchical decision tree (DT), Area Under Curve.**

I. INTRODUCTION

AD is a grievous neurodegenerative disease which is one of the chronic diseases that needs early detection, so that the treatment can be effective. Usually it starts slowly but with time worsens. This disease primarily affects the older people [1] and can become the cause for dementia [2]. The task of detecting this disease at the early stage is very difficult but equally important, so there is a necessity of intelligent system for supporting the clinicians in the early diagnosis of this disease. To address the mentioned problem, this paper elaborates the machine learning concept. Due to the handiness of improved technology, data exponentially increases and the world becomes a data rich society. This enormous amount of data takes the learning algorithms of machine at paramount height. Analyzing the immense data to get the fruitful result is the area of budding research. The target of all learning technologies is to retrieve the unseen patterns that can be further helpful in taking decision. Learning techniques are abundantly found in different sectors like media, health, agriculture, etc. Discovering and tackling of data is the herculean effort without intelligent learning models. Intelligent methods of learning are very useful for getting unseen patterns, trends, and relation between the features. The objective is to elaborate learning paradigms to International Journal of Advanced Science and Technology Vol. 29, No. 04, (2020), pp. 7062 – 7073 ISSN: 2005-4238 IJAST 7063 Copyright ⓒ 2019 SERSC model as well as to analyze the MRI data [1]. The vital role of learning methods in analyzing data motivates this research for comprehensive literature. As success of task directly depends on decision and correctness of decision depends on data as well as the proper analyses of data. This motivates the researchers either involved in academic or industry to work on techniques of learning models. Thus, this article revisits the present research and performed a systematic study. To know the potential of various well-known algorithms of machine learning such like LR, SVM, DT, RF, and boosting adaboost, the experiment conducted on all the mentioned learning approaches with the help of oasis_longitudinal MRI data. This dataset provides a useful aid for the diagnosis with the tracking of AD. The efficiency of the above mentioned models is measured in terms of accuracy, recall, and AUC (Area Under Curve). Additionally, time taken by the respective learning methodologies is also calculate

1.1 MOTIVATION

Due to the handiness of improved technology, data exponentially increases and the world becomes a data rich society. This enormous amount of data takes the learning

algorithms of machine at paramount height. Analyzing the immense data to get the fruitful result is the area of budding research. The target of all learning technologies is to retrieve the unseen patterns that can be further helpful in taking decision. Learning techniques are abundantly found in different sectors like media, health, agriculture, etc. Discovering and tackling of data is the herculean effort without intelligent learning models. Intelligent methods of learning are very useful for getting unseen patterns, trends, and relation between the features.

1.2 EXISTING SYSTEM

This task performs on network layers. This study aids the readers in knowing the protocols related to the DL-enhanced in wireless network.

1.2.1 Limitations of existing system

Raw data is elaborated. So, it takes more time for implementation

1.3 OBJECTIVES

The objective is to elaborate learning paradigms to model as well as to analyze the MRI data [1]. The vital role of learning methods in analyzing data motivates this research for comprehensive literature. As success of task directly depends on decision and correctness of decision depends on data as well as the proper analyses of data. This motivates the researchers either involved in academic or industry to work on techniques of learning models. Thus, this article revisits the present research and performed a systematic study

1.4 OUTCOMES

Random Forest and AdaBoost achieves high accuracy as compared with others. This is because AdaBoost has the capability to turn weak classifier to strong one. Random Forest strength to overcome the overfitting problem, makes it better than the others.

- Random Forest gets high recall or true positive rate due to reduction of overfitting problem.
- AUC is a performance measure parameter, which is high with Random Forest.
- Along with good performance, Random Forest classifier takes more time to execute. This classifier training speed is low as compared to others.

1.5 APPLICATIONS

This strategy used in **Diagnosis of Alzheimer Disease using Machine Learning Approaches**

1.6 STRUCTURE OF PROJECT (SYSTEM ANALYSIS)

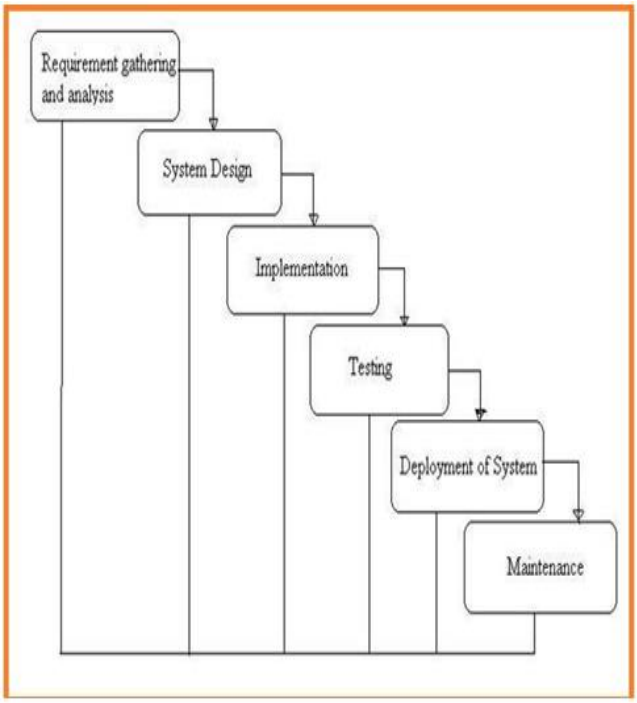


Fig: 1 Project SDLC

- Project Requisites Accumulating and Analysis
- Application System Design
- Practical Implementation
- Manual Testing of My Application
- Application Deployment of System
- Maintenance of the Project

1.6.1 REQUISITES ACCUMULATING AND ANALYSIS

It's the first and foremost stage of the any project as our is a an academic leave for requisites amassing we followed of IEEE Journals and Amassed so many IEEE Relegated papers and final culled a Paper designated "Individual web revisitation by setting and substance importance input and for analysis stage we took referees from the paper and did literature survey of some papers and amassed all the Requisites of the project in this stage

1.6.2 SYSTEM DESIGN

In System Design has divided into three types like GUI Designing, UML Designing with avails in development of project in facile way with different actor and its utilizer case by utilizer case diagram, flow of the project utilizing sequence, Class diagram gives information about different class in the project with methods that have to be utilized in the project if comes to our project our UML Will utilizable in this way The third and post import for the project in system design is Data base design where we endeavor to design database predicated on the number of modules in our project

1.6.3 IMPLEMENTATION

The Implementation is Phase where we endeavor to give the practical output of the work done in designing stage and most of Coding in Business logic lay coms into action in this stage its main and crucial part of the project

1.6.4 TESTING UNIT TESTING

It is done by the developer itself in every stage of the project and fine-tuning the bug and module predicated additionally done by the developer only here we are going to solve all the runtime errors
MANUAL TESTING
As our Project is academic Leave, we can do any automatic testing so we follow manual testing by endeavor and error methods

1.6.5 DEPLOYMENT OF SYSTEM AND MAINTENANCE

Once the project is total yare, we will come to deployment of client system in genuinely world as its academic leave we did deployment i our college lab only with all need Software's with having Windows OS .
The Maintenance of our Project is one-time process only.

1.7 FUNCTIONAL REQUIREMENTS

1. Data Collection
2. Data Preprocessing
3. Training And Testing
4. Modeling
5. Predicting

1.8 NON FUNCTIONAL REQUIREMENTS

NON-FUNCTIONAL REQUIREMENT (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system. Example of nonfunctional requirement, *"how fast does the website load?"* Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non-functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users are > 10000. Description of non-functional requirements is just as critical as a functional requirement.

- Usability requirement
- Serviceability requirement
- Manageability requirement
- Recoverability requirement
- Security requirement
- Data Integrity requirement
- Capacity requirement
- Availability requirement
- Scalability requirement
- Interoperability requirement
- Reliability requirement
- Maintainability requirement
- Regulatory requirement
- Environmental requirement

1.8.1 EXAMPLES OF NON-FUNCTIONAL REQUIREMENTS

Here, are some examples of non-functional requirement:

- Users must upload dataset
- The software should be portable. So moving from one OS to other OS does not create any problem.
- Privacy of information, the export of restricted technologies, intellectual property rights, etc. should be audited.

1.8.2 ADVANTAGES OF NON-FUNCTIONAL REQUIREMENT

Benefits/pros of Non-functional testing are:

- The nonfunctional requirements ensure the software system follow legal and compliance rules.
- They ensure the reliability, availability, and performance of the software system
- They ensure good user experience and ease of operating the software.
- They help in formulating security policy of the software system.

1.8.3 DISADVANTAGES OF NON-FUNCTIONAL REQUIREMENT

Cons/drawbacks of Non-function requirement are:

- None functional requirement may affect the various high-level software subsystem
- They require special consideration during the software architecture/high-level design phase which increases costs.
- Their implementation does not usually map to the specific software sub-system,
- It is tough to modify non-functional once you pass the architecture phase.

1.8.4 KEY LEARNING

The character of the time period, the length of road, the weather, the bus speed and the rate of road usage are adopted as input vectors in Support Vector Machine.

2. PROBLEM IDENTIFICATION AND OBJECTIVES

2.1 EXISTING APPROACH

- This task performs on network layers. This study aids the readers in knowing the protocols related to the DL-enhanced in wireless network.

- **Drawbacks**

Raw data is elaborated. So , it takes more time for implementation

2.2 PROPOSED SYSTEM

- To cope up this problem, this paper elaborates and tests various approaches of machine learning for AD diagnosis. Using oasis longitudinal MRI data, this study train various models of machine learning to detect patients suffering from the AD. Along with the machine learning, this letter addressed the deep learning comprehensive overview. With various learning approaches such like binary classifier named logistic regression (LR), support vector machine (SVM), hierarchical decision tree (DT), ensemble random forest (RF), and boosting adaboost, experimental result is analyzed in terms of accuracy, recall, and AUC (Area Under Curve).

2.2.1 ADVANTAGES

- Where feasible, networks of specialist diagnostic centres should be established to confirm early stage dementia diagnoses and formulate care management plans.

2.3 SOFTWARE AND HARDWARE REQUIREMENTS

2.3.1 SOFTWARE REQUIREMENTS

The functional requirements or the overall description documents include the product perspective and features, operating system and operating environment, graphics requirements, design constraints and user documentation.

The appropriation of requirements and implementation constraints gives the general overview of the project in regards to what the areas of strength and deficit are and how to tackle them.

- Python idel 3.7 version (or)
- Anaconda 3.7 (or)
- Jupiter (or)
- Google colab

2.3.2 HARDWARE REQUIREMENTS

Minimum hardware requirements are very dependent on the particular software being developed by a given a thought Python / Canopy / VS Code user. Applications that need to store large arrays/objects in memory will require more RAM, whereas applications that need to perform numerous calculations or tasks more quickly will require a faster processor.

- Operating system : windows, linux
- Processor : minimum intel i3
- Ram : minimum 4 gb
- Hard disk : minimum 250gb

2.4 ABOUT DATASET

Alzheimer's Disease prediction:

For alzheimer’s disease prediction we take patients details. This dataset is downloaded from kaggle datasets.

For this prediction we uses oasis longitudinal dataset. This dataset consists of fifteen columns and 373 records. Out of fifteen columns, fourteen are features and one column represents label. Labels are classified into three categories, named are Nondemented, Demented, and Converted.

2.5 ALGORITHMS

Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make predictions for each tree created in the first phase.

The Working process can be explained in the below steps and diagram:

Step-1: Select random K data points from the training set.

Step-2: Build the decision trees associated with the selected data points (Subsets).

Step-3: Choose the number N for decision trees that you want to build.

Step-4: Repeat Step 1 & 2.

Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

- Logistic Regression, Support Vector Machine, Decision Tree, Random Forest, Ada boost

3. IMPLEMENTATION

3.1 FLOW CHART:

4. RESULTS AND DISCUSSIONS

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
df_cross = pd.read_csv('data/oasis_cross-sectional.csv')
```

```
df_long = pd.read_csv('data/oasis_longitudinal.csv')
```

```
df_cross.head()
```

	ID	M/F	Hand	Age	Educ	SES	MMSE	CDR	eTIV	nWBV	ASF	Delay
0	OAS1_0001_MR1	F	R	74	2.0	3.0	29.0	0.0	1344	0.743	1.306	NaN
1	OAS1_0002_MR1	F	R	55	4.0	1.0	29.0	0.0	1147	0.810	1.531	NaN
2	OAS1_0003_MR1	F	R	73	4.0	3.0	27.0	0.5	1454	0.708	1.207	NaN
3	OAS1_0004_MR1	M	R	28	NaN	NaN	NaN	NaN	1588	0.803	1.105	NaN
4	OAS1_0005_MR1	M	R	18	NaN	NaN	NaN	NaN	1737	0.848	1.010	NaN

Importing the required package and dataset which is available in kaggle site for alzheimer disease using ML Filling Nan values with mean of other values in dataset

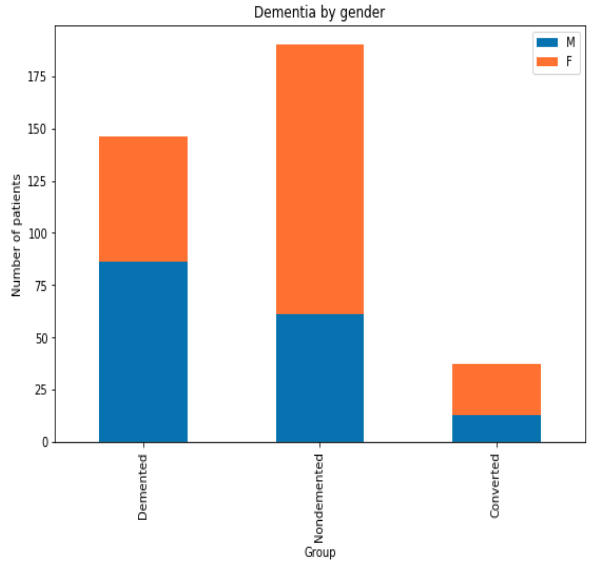
Visualization of outcomes of based Gender

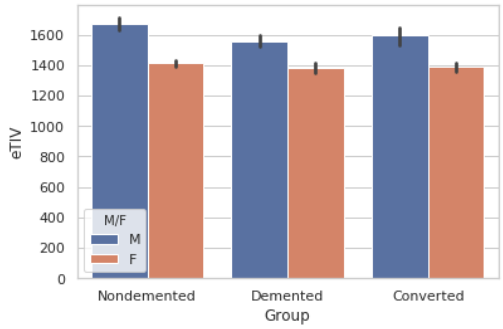
```
df_cross['Educ'].fillna((round(df_cross['Educ'].mean())), inplace=True)
df_cross['SES'].fillna((round(df_cross['SES'].mean())), inplace=True)
df_cross['MMSE'].fillna((round(df_cross['MMSE'].mean())), inplace=True)
df_cross['CDR'].fillna((round(df_cross['CDR'].mean())), inplace=True)
df_cross['Delay'].fillna((round(df_cross['Delay'].mean())), inplace=True)
```

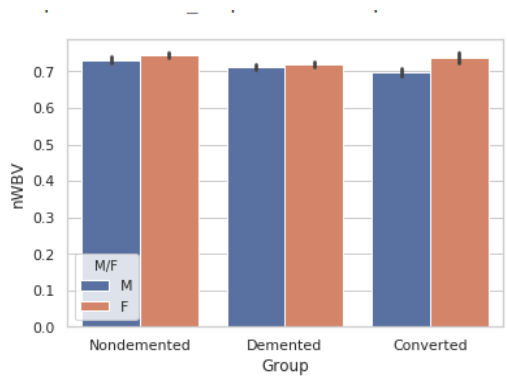
```
df_cross.head()
```

	ID	M/F	Hand	Age	Educ	SES	MMSE	CDR	eTIV	nWBV	ASF	Delay
0	OAS1_0001_MR1	F	R	74	2.0	3.0	29.0	0.0	1344	0.743	1.306	21.0
1	OAS1_0002_MR1	F	R	55	4.0	1.0	29.0	0.0	1147	0.810	1.531	21.0
2	OAS1_0003_MR1	F	R	73	4.0	3.0	27.0	0.5	1454	0.708	1.207	21.0
3	OAS1_0004_MR1	M	R	28	3.0	2.0	27.0	0.0	1588	0.803	1.105	21.0
4	OAS1_0005_MR1	M	R	18	3.0	2.0	27.0	0.0	1737	0.848	1.010	21.0

```
Text(0.5, 1.0, 'Dementia by gender')
```







Visualizing the Outcome based on other columns data

Dataset Main.csv has been modified dataset from previous one, where groups label was replaced from Dementia, non-dementia and converted to 0,1,2.

```
import pandas as pd
df_long = pd.read_csv('data/main.csv')
df_long.head()
```

	Group	Sex	Age	EDUC	SES	MMSE	CDR	eTIV	nWBV	ASF
0	0	1	87	14	2	27	0.0	1987	0.696	0.883
1	1	1	88	14	2	30	0.0	2004	0.681	0.876
2	1	1	75	12	3	23	0.5	1678	0.736	1.046
3	1	1	76	12	4	28	0.5	1738	0.713	1.010
4	1	1	80	12	3	22	0.5	1698	0.701	1.034

```
#array = df_long.values
X = df_long.drop("Group",axis=1)
#X = array[:,[3, 6, 8, 9]] #Feature Extraction with RFE
#Y_float64 = array[:,12] # all rows and CDR column used for AUC ROC calculations late
#Y = array[:,2] # all rows and CDR_Class column
#Y_float64 = array[:,7] # all rows and CDR column used for AUC ROC calculations later
Y=df_long["Group"]
validation_size = 0.20
seed = 123456
X_train, X_validation, Y_train, Y_validation = train_test_split(X, Y,
test_size=validation_size, random_state=seed)
```

Data has been split into training and testing and preprocessed for modelling.
Deploying the ML

Support Vector Machine Classifier

```
from sklearn.svm import SVC
SVM = SVC(kernel='linear')
SVM.fit(X_train, Y_train)
predictions = SVM.predict(X_validation)
val1 = (accuracy_score(Y_validation, predictions)*100)
print("*Accuracy score for SVM: ", val1, "\n")
print("*Confusion Matrix for SVM: ")
print(confusion_matrix(Y_validation, predictions))
print("*Classification Report for SVM: ")
print(classification_report(Y_validation, predictions))
```

```
*Accuracy score for SVM:  90.66666666666666

*Confusion Matrix for SVM:
[[39  0  0]
 [ 0 27  3]
 [ 3  1  2]]
*Classification Report for SVM:
              precision    recall  f1-score   support

           0       0.93      1.00      0.96        39
           1       0.96      0.90      0.93        30
           2       0.40      0.33      0.36         6

    accuracy                           0.91        75
   macro avg       0.76      0.74      0.75        75
```

Decision Tree Classifier

```
from sklearn import tree
DT = tree.DecisionTreeClassifier()
DT.fit(X_train, Y_train)
predictions = DT.predict(X_validation)
val2 = (accuracy_score(Y_validation, predictions)*100)
print("*Accuracy score for DT: ", val2, "\n")
print("*Confusion Matrix for DT: ")
print(confusion_matrix(Y_validation, predictions))
print("*Classification Report for DT: ")
print(classification_report(Y_validation, predictions))
```

```
*Accuracy score for DT:  86.66666666666667

*Confusion Matrix for DT:
[[38  0  1]
 [ 0 26  4]
 [ 3  2  1]]
*Classification Report for DT:
              precision    recall  f1-score   support

           0       0.93      0.97      0.95        39
           1       0.93      0.87      0.90        30
           2       0.17      0.17      0.17         6

    accuracy                           0.87        75
   macro avg       0.67      0.67      0.67        75
weighted avg       0.87      0.87      0.87        75
```

Random Forest

```
from sklearn.ensemble import RandomForestClassifier
RF = RandomForestClassifier()
RF.fit(X_train, y_train)
predictions = RF.predict(X_test)
val3 = (accuracy_score(y_test, predictions)*100)
print("*Accuracy score for RF: ", val3, "\n")
print("*Confusion Matrix for RF: ")
print(confusion_matrix(y_test, predictions))
print("*Classification Report for RF: ")
print(classification_report(y_test, predictions))
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, predictions)
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, predictions)
print(cm)

plt.matshow(cm)
plt.title('Confusion matrix of the classifier\n')
plt.xlabel('Predicted')
plt.ylabel('True')
plt.colorbar()
plt.show()
```

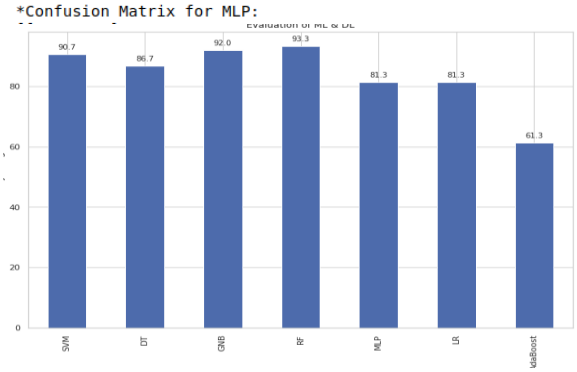
```
*Accuracy score for RF:  79.03225806451613

*Confusion Matrix for RF:
```

Model Comparsion

MLP

```
from sklearn.neural_network import MLPClassifier
MLP = MLPClassifier()
MLP.fit(X_train, y_train)
predictions = MLP.predict(X_test)
val4 = (accuracy_score(y_test, predictions)*100)
print("*Accuracy score for MLP: ", val4, "\n")
print("*Confusion Matrix for MLP: ")
print(confusion_matrix(y_test, predictions))
print("*Classification Report for MLP: ")
print(classification_report(y_test, predictions))

from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, predictions)
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, predictions)
print(cm)

plt.matshow(cm)
plt.title('Confusion matrix of the classifier\n')
plt.xlabel('Predicted')
plt.ylabel('True')
plt.colorbar()
plt.show()
```

```
*Accuracy score for MLP:  83.87096774193549

*Confusion Matrix for MLP:
```

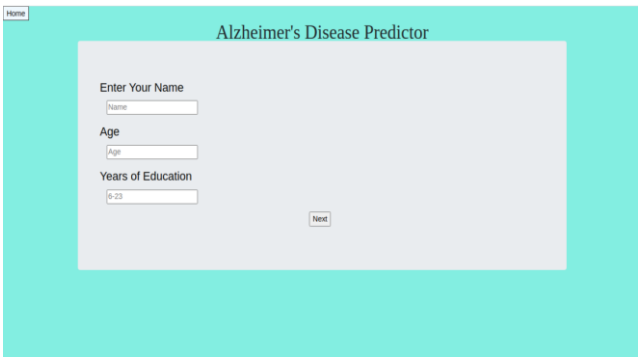


From the above comparison we can see RF giving better accurate comparision with other , So we are going to use RF for predicting the user input in flask framework.

```
    #result = "You suffer from a lifestyle disease! "
    #return jsonify(result=result)


@app.route("/home")
def home():
    return render_template('one.html')

if __name__ == '__main__':
    from werkzeug.serving import run_simple
    run_simple('localhost', 5000, app)
    #app.run(host='localhost',port=5000, debug=True,threaded=True)
```

```
 * Running on http://localhost:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [16/Oct/2020 10:01:01] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [16/Oct/2020 10:01:13] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [16/Oct/2020 10:01:51] "GET /one_set?name=Ramesh&age=87&yoe=14 HTTP/1.1" 200 -
127.0.0.1 - - [16/Oct/2020 10:02:05] "GET /two_set?ses=2&mmse=27&cdr=0.0 HTTP/1.1" 200 -
127.0.0.1 - - [16/Oct/2020 10:02:21] "GET /third_set?etiv=1987&nwbv=0.696&asf=0.883&gender=0 HTTP/1.1" 2

1

127.0.0.1 - - [16/Oct/2020 10:02:29] "GET /home HTTP/1.1" 200 -
```

Localhost



- Give next and enter the values on second and third page
- In third one after click next output will be displayed
- After entering the input we will get the status of patient

5. CONCLUSION

The paper presents the comprehensive overview of AD and how various learning approaches can analyze them. With the advancement in computational technology, data is enhancing day by day. It becomes difficult to handle this bulk of data. Machines as well as deep learning models explained in this study are the tools to tackle and analyze this bulky data. These learning models are able to analyze the data and can further classify or predict the result. Different learning model's analysis through experimental work are also shown in this article. The potential in terms of accuracy, recall, AUC, and time requirement to execute is analyzed efficiently in tabular as well as graphical form. With the comparative study of five ML models on oasis_longitudinal datasets, this paper concludes:

• Random Forest and AdaBoost achieves high accuracy as compared with others. This is because AdaBoost has the capability to turn weak classifier to strong one. Random Forest strength to overcome the overfitting problem, makes it better than the others.

• Random Forest gets high recall or true positive rate due to reduction of overfitting problem.

• AUC is a performance measure parameter, which is high with Random Forest.

• Along with good performance, Random Forest classifier takes more time to execute. This classifier training speed is low as compared to others

FUTURE WORK

Along with the overview of machine and deep learning models, this research gives data pre-processing details. This is performed by gathering information by surveying various papers. In the coming time, this research can be very

fruitful and utilized in applying learning models in different areas like health, agriculture, banking, etc. For analyzing the data, these learning models are great achievements for the scientists involved in both academic and industry. This article is useful for the researchers who are working in this direction and further can come up with more yielding outcomes.

REFERENCES

[1] J. Escudero, E. Ifeachor, J.P. Zajicek, C. Green, J. Shearer, S. Pearson, “Machine learning-based method for personalized and cost-effective detection of Alzheimer's disease”, IEEE transactions on biomedical engineering, Vol. 60, no. 1, (2013), pp. 164-168.

[2] Q. Zhou, M. Goryawala, Cabrerizo, J. Wang, W. Barker, D.A. Loewenstein, R. Duara, M. Adjouadi, “An optimal decisional space for the classification of Alzheimer's disease and mild cognitive impairment”, IEEE Transactions on Biomedical Engineering, Vol. 61, no. 8, (2014), pp. 2245-2253.

[3] S. Athmaja, M. Hanumanthappa, V. Kavitha, “A survey of machine learning algorithms for big data analytics”, Proceedings of the IEEE International Conference on Innovations in Information, Embedded and Communication Systems, Coimbatore, India, (2017) March 17-18.

[4] M. Bkassiny, Y. Li, S.K. Jayaweera, “A survey on machine-learning techniques in cognitive radios”, IEEE Communications Surveys & Tutorials, Vol. 15, no. 3, (2012), pp. 1136-1159.

[5] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, V. C. M. Leung, "A Survey on Security Threats and Defensive Techniques of Machine Learning: A Data Driven View," IEEE Access, vol. 6, (2018), pp. 12103- 12117.

[6] Y. Ji, Q. Wang, X. Li, J. Liu, “A Survey on Tensor Techniques and Applications in Machine Learning” IEEE Access, Vol. 7, (2019), pp. 162950-162990.

[7] S. Das, M.J. Nene, “A survey on types of machine learning techniques in intrusion prevention systems”, Proceedings of IEEE International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), Chennai, India, (2017) March 22-24.

[8] P. R. Hegde, M. M. Shenoy, B. H. Shekar, "Comparison of Machine Learning Algorithms for Skin Disease Classification Using Color and Texture Features", Proceedings of IEEE International Conference on Advances in Computing, Communications and Informatics, Bangalore, India, (2018) Sep 19-22.

[9] Q. Mao, F. Hu, Q. Hao, “Deep learning for intelligent wireless networks: A comprehensive survey”, IEEE Communications Surveys & Tutorials, Vol. 20, no. 4, (2018), pp. 2595-2621.