THYROID DISEASE DETECTION

OJJASWI NIRMAL
KALPANA GAUNOLLA

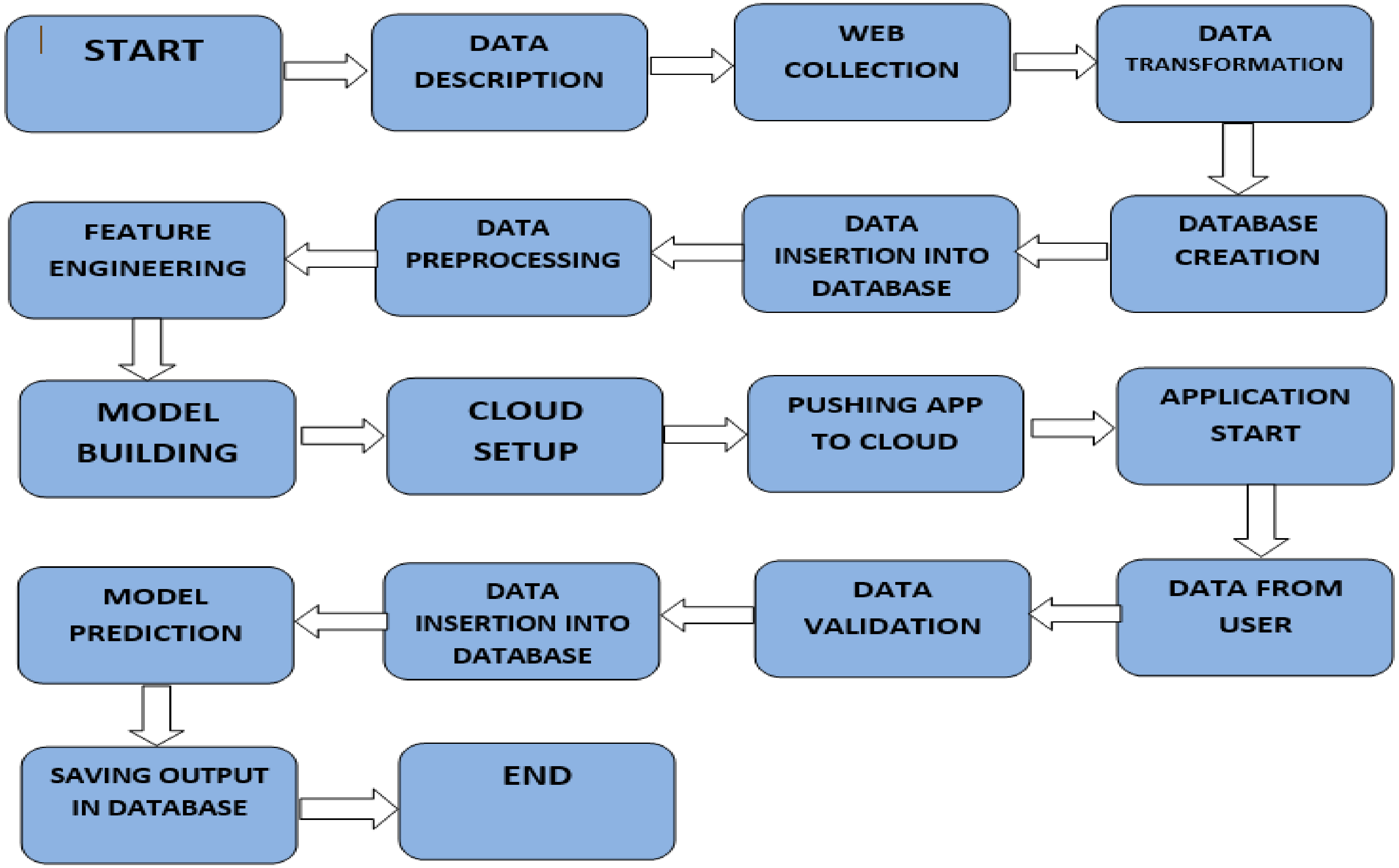
INTRODUCTION

Thyroid disease a very common problem in India, more than one crore people are suffering with the disease every year. Especially it is more common in female. Hyperthyroidism and hypothyroidism are the most two common diseases caused by irregular function of thyroid gland. Thyroid disorder can speed up or slow down the metabolism of the body. In the world of rising new technology and innovation, healthcare industry is advancing with the role of Artificial Intelligence. Machine learning algorithms can help to early detection of the disease and to improve the quality of the life. This study demonstrates the how different classification algorithms can forecasts the presence of the disease. Different classification algorithms such as Logistic regression, Random Forest, Decision Tree, Naïve Bayes, Support Vector Machine have been tested and compared to predict the better outcome of the model.

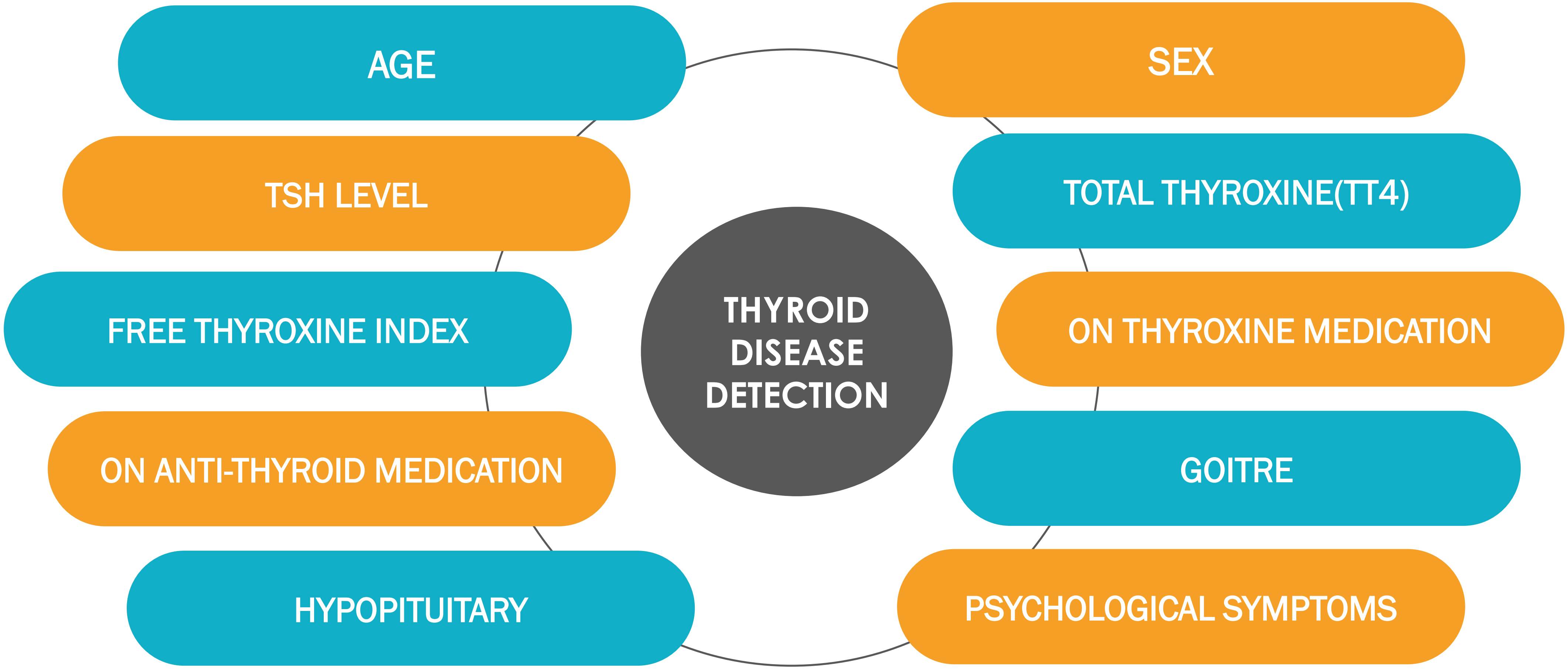
OBJECTIVE

The main goal of this project is to predict the risk of hyperthyroid and hypothyroid based on various factors of individuals. Thyroid disease is a common cause of medical diagnosis and prediction, with an onset that is difficult to forecast in medical research. It will play a decisive role in order to early detection, accurate identification of the disease and helps the doctors to make proper decisions and better treatment.

ARCHITECTURE



DATASET



THYROID DISEASE DETECTION

AGE

TSH LEVEL

FREE THYROXINE INDEX

ON ANTI-THYROID MEDICATION

HYPOPITUITARY

SEX

TOTAL THYROXINE(TT4)

ON THYROXINE MEDICATION

GOITRE

PSYCHOLOGICAL SYMPTOMS

DATA ANALYSIS STEPS

DATA COLLECTION

In step 1, we collect data which is generally present in a database or on internet.

DATA PREPROCESSING

In step 2, we preprocess the data which involves data cleaning by handling outliers, null values etc.

EXPLORATORY DATA ANALYSIS

In step 3, we explore the data by performing univariate and bivariate analysis on the features.

FEATURE SELECTION

In step 4, we use feature selection techniques to filter out the most important features to perform model creation

MODEL CREATION AND EVALUATION

In step 5, we finally build models on our dataset and choose the model which gives the best accuracy.

RANDOM FOREST MODEL

INTRODUCTION

❑ The random forest classifier is a supervised learning algorithm which we can use for regression and classification problems. It is among the most popular machine learning algorithms due to its high flexibility and ease of implementation.

❑ It is called Random Forest because it consists of multiple decision trees just as a forest has many trees. On top of that, it uses randomness to enhance its accuracy and combat overfitting, which can be a huge issue for such a sophisticated algorithm. These algorithms make decision trees based on a random selection of data samples and get predictions from every tree. After that, they select the best viable solution through votes.

❑ Random Forest Classifier being ensembled algorithm tends to give more accurate result. This is because it works on the principle i.e. number of weak estimators when combined forms strong estimator. Even if one or few decision trees are prone to noise, overall results would tend to be correct.
Even with small number of estimators (=30), its gives us high accuracy as 97%.

RANDOM FOREST MODEL

ADVANTAGES

- The random forest algorithm is significantly more accurate than most of the non-linear classifiers.
- This algorithm is also very robust because it uses multiple decision trees to arrive at its result.
- The random forest classifier doesn’t face the overfitting issue because it takes the average of all predictions, canceling out the biases and thus, fixing the overfitting problem.
- Random forests don’t let missing values cause an issue. They can use median values to replace the continuous variables or calculate the proximity-weighted average of the missing values to solve this problem.
- This algorithm offers you relative feature importance that allows you to select the most contributing features for your classifier easily.

RANDOM FOREST MODEL

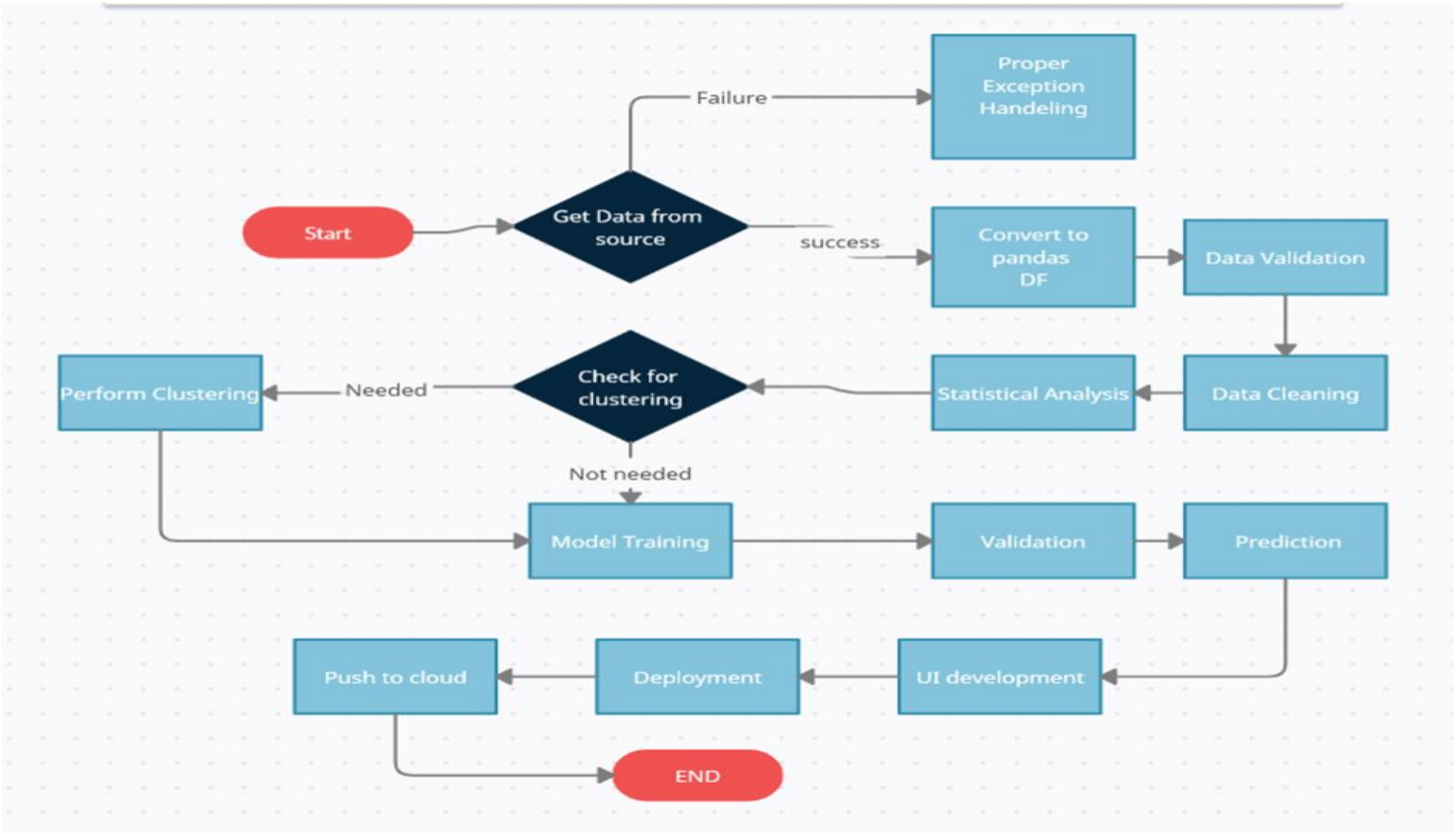
DISADVANTAGES

- This algorithm is substantially slower than other classification algorithms because it uses multiple decision trees to make predictions. When a random forest classifier makes a prediction, every tree in the forest has to make a prediction for the same input and vote on the same. This process can be very time-consuming.

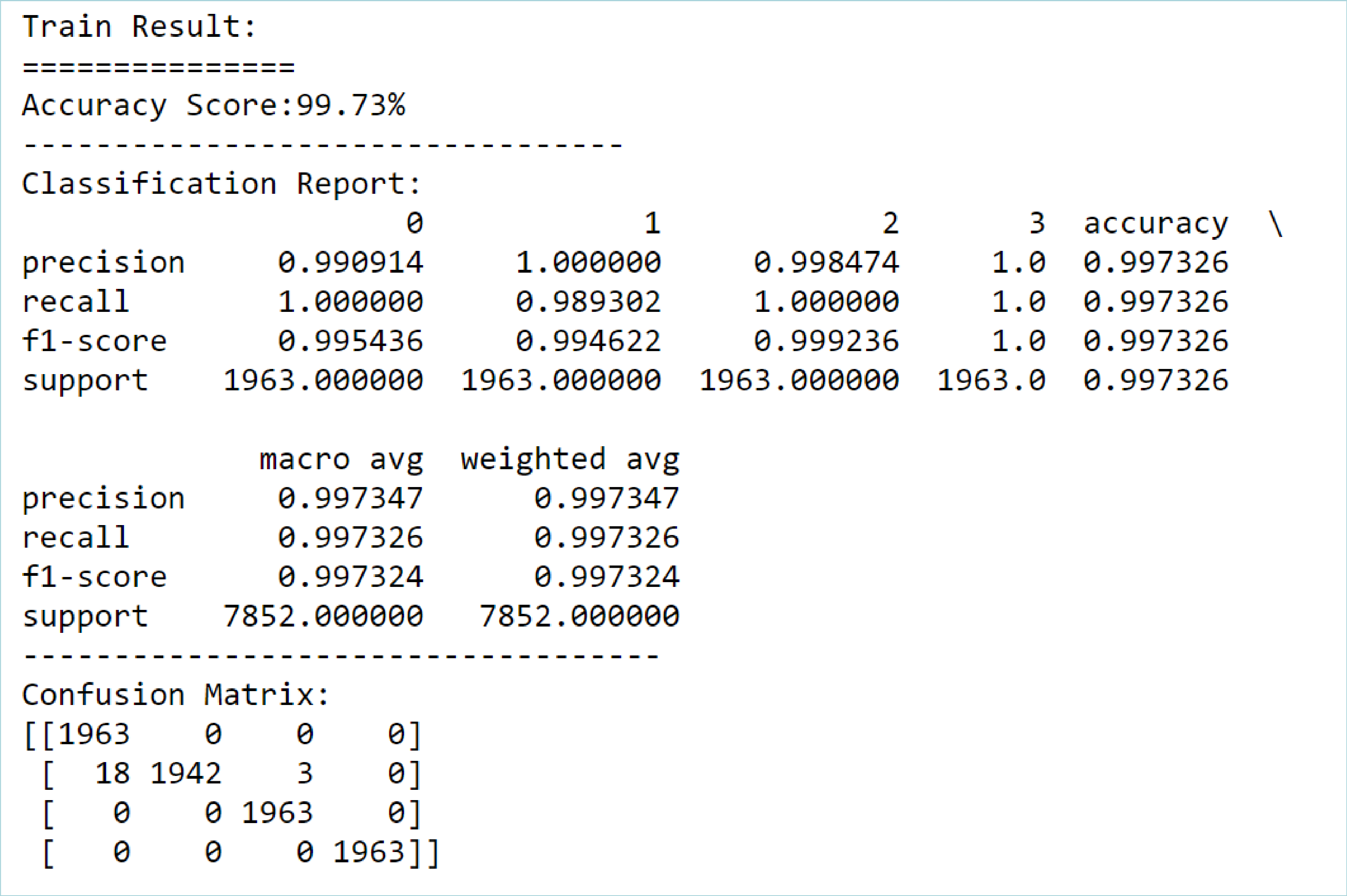
- Because of its slow pace, random forest classifiers can be unsuitable for real-time predictions.

- The model can be quite challenging to interpret in comparison to a decision tree as you can make a selection by following the tree’s path. However, that’s not possible in a random forest as it has multiple decision trees.

MODEL TRAINING AND VALIDATION WORKFLOW



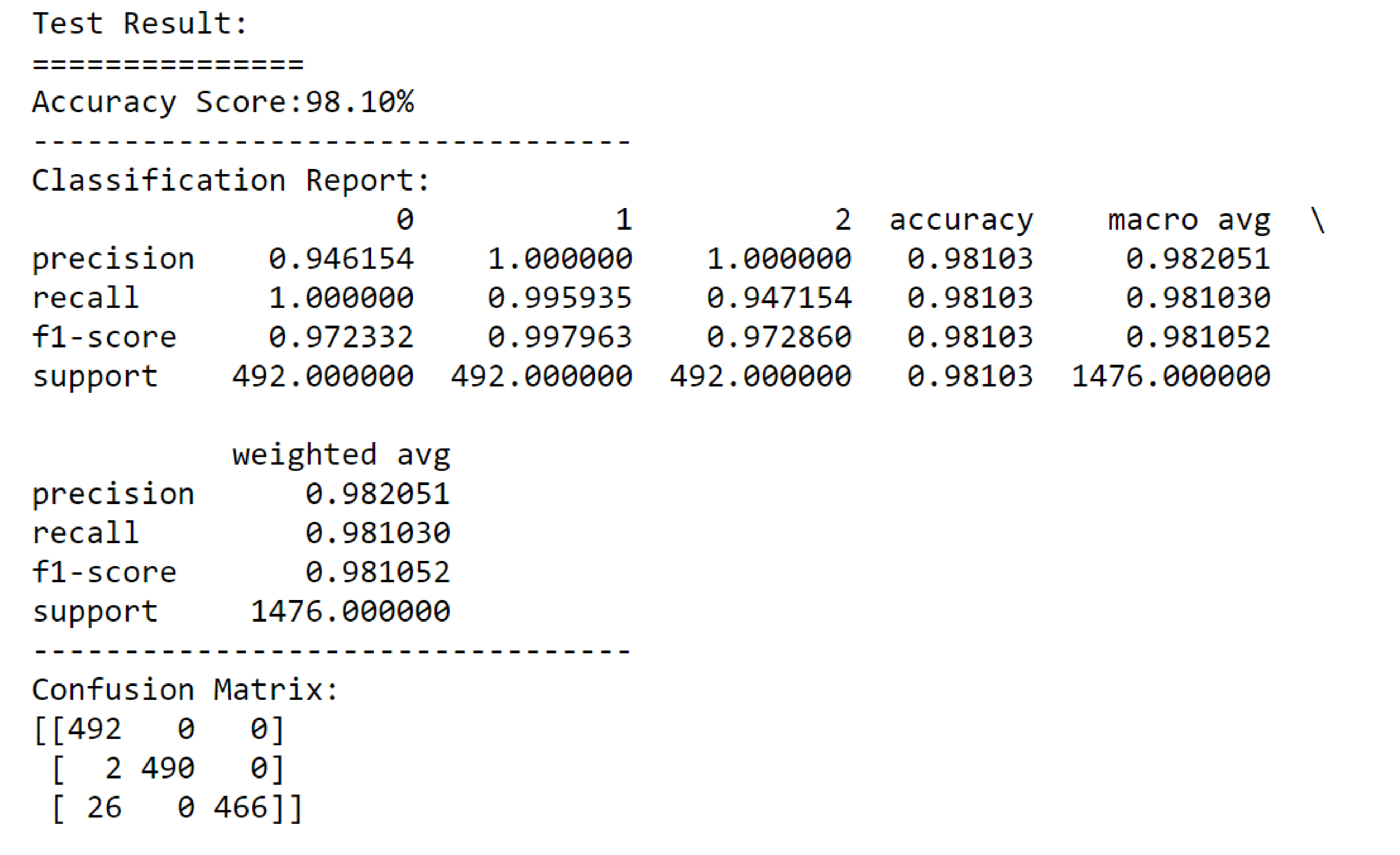
MODEL PREDICTION RESULTS ON TRAINING DATASET



```
Train Result:
===============
Accuracy Score:99.73%
_________________________________
Classification Report:
                     0            1            2       3  accuracy  \
precision     0.990914     1.000000     0.998474     1.0  0.997326
recall        1.000000     0.989302     1.000000     1.0  0.997326
f1-score      0.995436     0.994622     0.999236     1.0  0.997326
support    1963.000000  1963.000000  1963.000000  1963.0  0.997326

             macro avg  weighted avg
precision     0.997347      0.997347
recall        0.997326      0.997326
f1-score      0.997324      0.997324
support    7852.000000   7852.000000
___________________________________
Confusion Matrix:
[[1963    0    0    0]
 [  18 1942    3    0]
 [   0    0 1963    0]
 [   0    0    0 1963]]
```

MODEL PREDICTION RESULTS ON TEST DATASET



```
Test Result:
===============
Accuracy Score:98.10%
_________________________________
Classification Report:
                    0           1           2  accuracy     macro avg  \
precision    0.946154    1.000000    1.000000   0.98103      0.982051
recall       1.000000    0.995935    0.947154   0.98103      0.981030
f1-score     0.972332    0.997963    0.972860   0.98103      0.981052
support    492.000000  492.000000  492.000000   0.98103   1476.000000

           weighted avg
precision      0.982051
recall         0.981030
f1-score       0.981052
support     1476.000000
_________________________________
Confusion Matrix:
[[492   0   0]
 [  2 490   0]
 [ 26   0 466]]
```

THANK YOU