




Thyroid Disease Detection Using Machine Learning

-By : Rushikesh Bodhe


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.

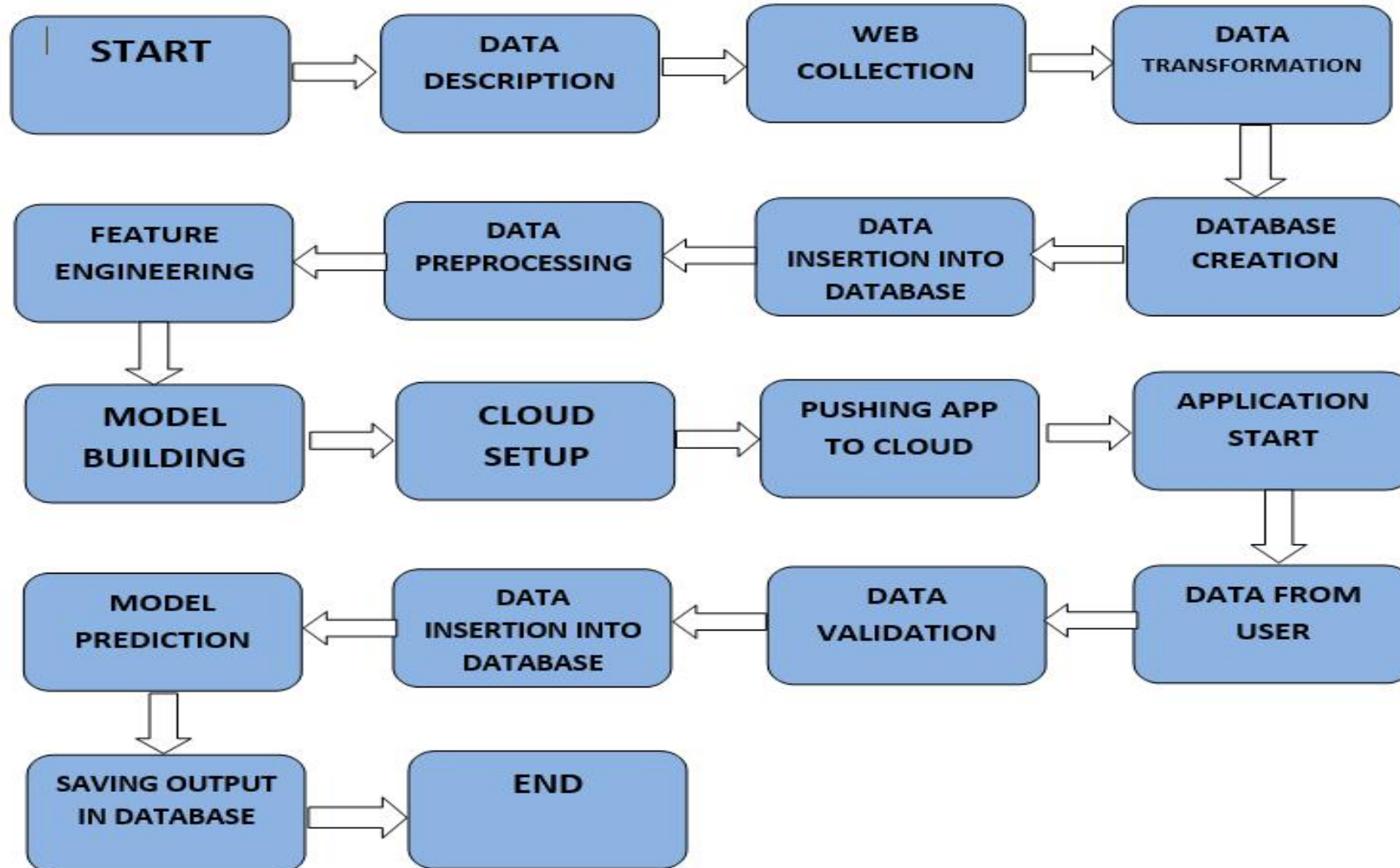
OBJECTIVE



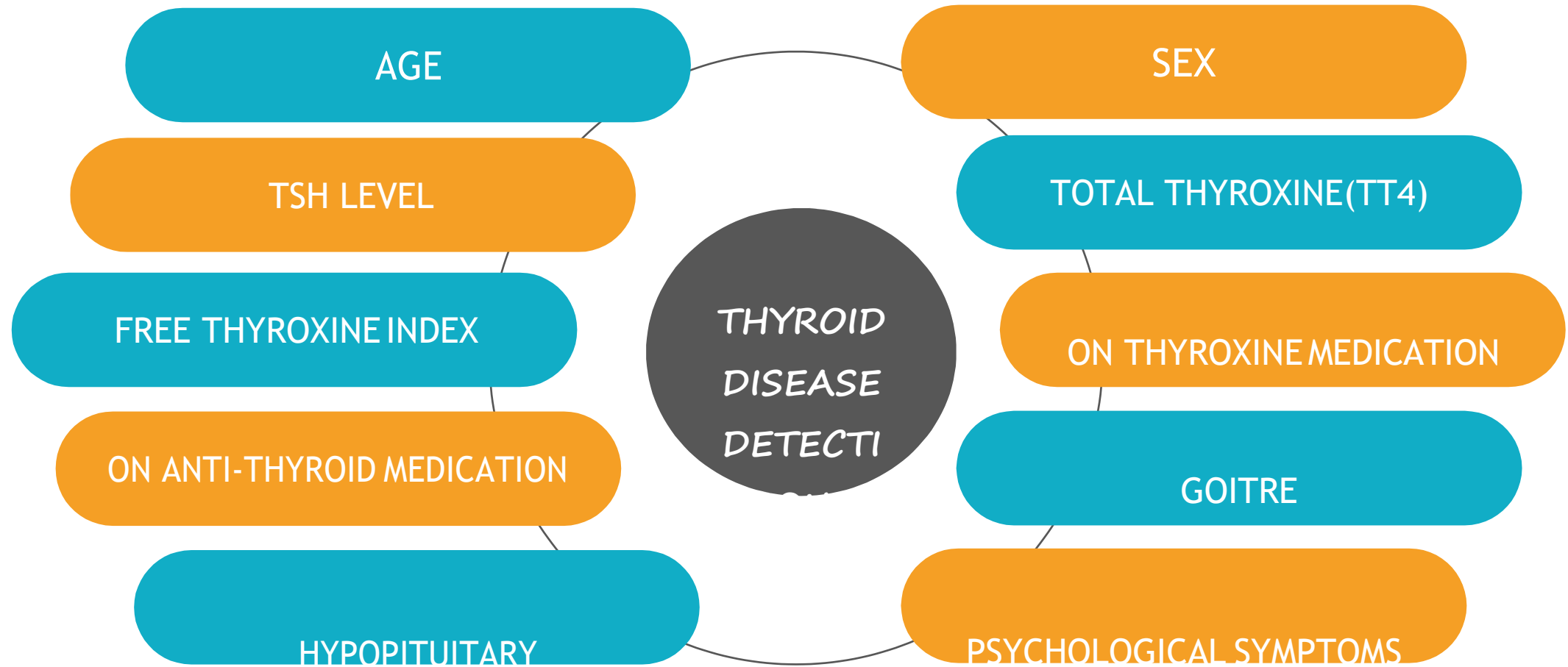
The main goal of this project is to predict the risk of hyperthyroid and hypothyroid based on various factors of individuals. Thyroid disease is a common cause of medical diagnosis and prediction, with an onset that is difficult to forecast in medical research. It will play a decisive role in order to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



ARCHITECTURE



DATASET



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.

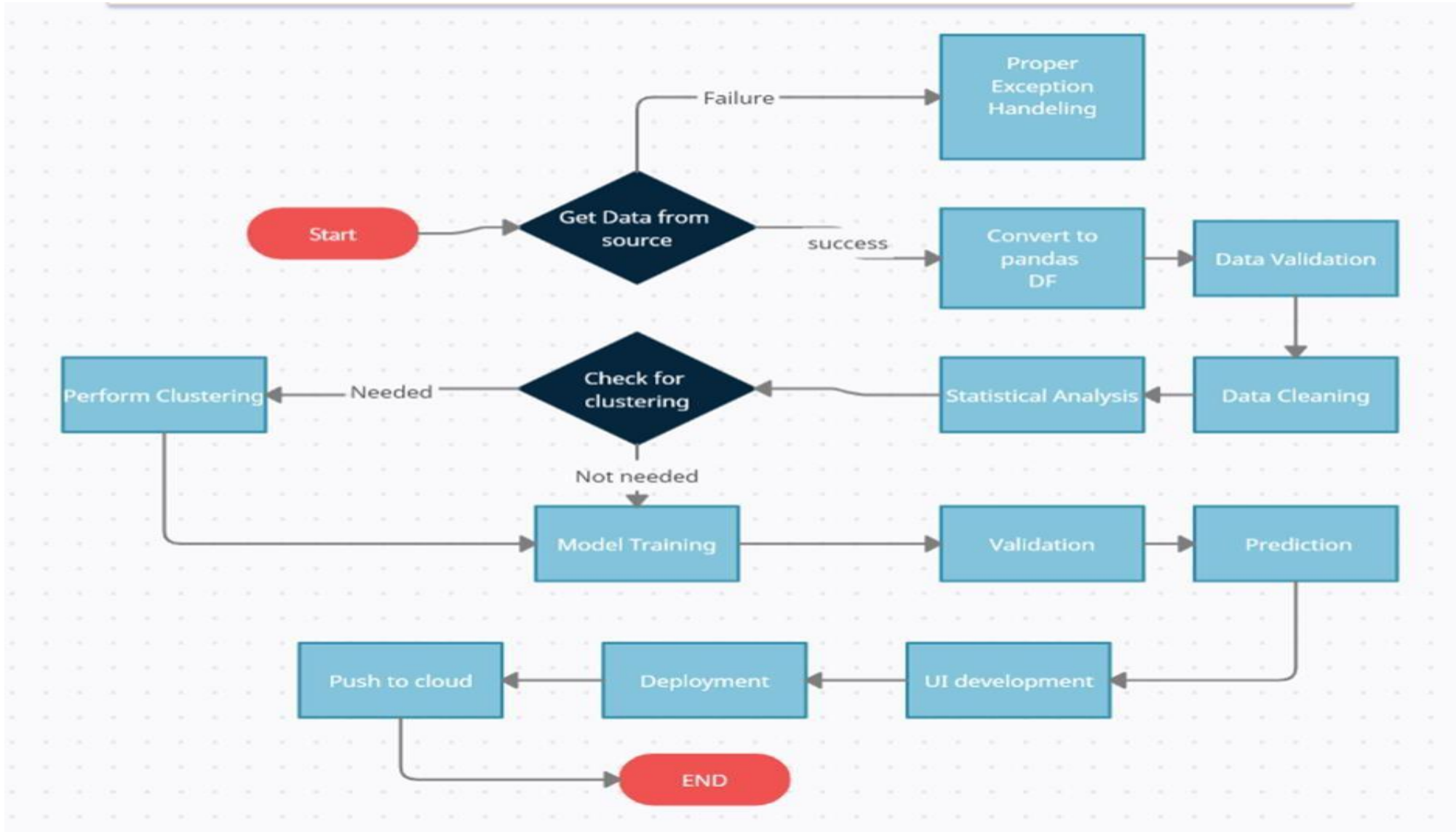
RANDOM FOREST MODEL

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 an ensemble algorithm tends to give more accurate results. This is because it works on the principle i.e. number of weak estimators when combined forms a strong estimator. Even if one or few decision trees are prone to noise, overall results would tend to be correct.

Even with a small number of estimators ($n=30$), it gives us high accuracy as 97%.

MODEL TRAINING AND VALIDATION WORKFLOW



MODEL TRAINING AND VALIDATION WORKFLOW



Data Collection

- ThyroidDiseaseDataSetfromUCIMachineLearningRepository

Data Pre-processing

- Missing values handling by Simple imputation (median strategy)
- Outliers' detection and removal by boxplot and percentile methods
- Categorical features handling by ordinal encoding and label encoding
- Feature scaling done by Standard Scalar method
- Imbalanced dataset handled by SMOTE - Over sampling
- Feature selection done by forward feature selection

MODEL TRAINING AND VALIDATION WORKFLOW

Model Creation and Evaluation

- ❑ Various classification algorithms like Logistic Regression, Random Forest, Decision Tree, Naïve Bayes, Support Vector Machine tested.
- ❑ Random Forest, Decision Tree and Logistic regression were given better results. Random Forest was chosen for the final model training and testing.
- ❑ Hyper parameter tuning was performed.
- ❑ Model performance evaluated based on accuracy, confusion matrix, classification report.

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]]
```

DATABASE CONNECTION & DEPLOYMENT

Database Connection

- ☐ MongoDB Atlas database used for this project

Model Deployment

- ☐ The final model is deployed using on Heroku using Flask framework





THANK YOU