High Level Design
Thyroid Disease Prediction

Revision Number: 1.0
Last date of revision: 12/1/2023

Dhaksin S

Document Version Control

Date Issued	Version	Description	Author
10 Jan 2023	1.0	First Version of Complete HLD	Dhaksin S

Contents

Abstract

This study explores machine learning for thyroid disease prediction using clinical and laboratory data. Preprocessing includes handling missing values and encoding categorical variables. Exploratory data analysis reveals feature relationships. Principal Component Analysis (PCA) reduces dimensionality,

and MinMaxScaler scales features. Models like Decision Tree, Random Forest, KNN, SVM, and Logistic Regression are trained and tested, with Random Forest showing the highest accuracy. The study underscores machine learning's role in early thyroid disease detection and treatment.

1. Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like:

  ○ Security ○ Reliability ○
  Maintainability ○
  Portability ○ Reusability ○
  Application compatibility ○
  Resource utilization ○
  Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

2. General Description

2.1 Product Perspective & Problem Statement

The thyroid disease prediction system integrates machine learning for early detection and management. It offers a user-friendly interface for healthcare professionals, aiming to improve diagnostic accuracy and patient outcomes.

Current methods for thyroid disorder diagnosis are often inaccurate and inefficient, leading to delayed treatment. There is a need for an automated system that can accurately predict thyroid disease based on patient data to enable timely intervention.

2.2 Tools and Libraries used

The code uses pandas for data manipulation, numpy for numerical operations, matplotlib.pyplot and seaborn for data visualization, and scikit-learn for machine learning tasks. It also utilizes KNNImputer for missing value imputation and LabelEncoder for encoding categorical variables.

3. Design Details

3.1 Functional Architecture

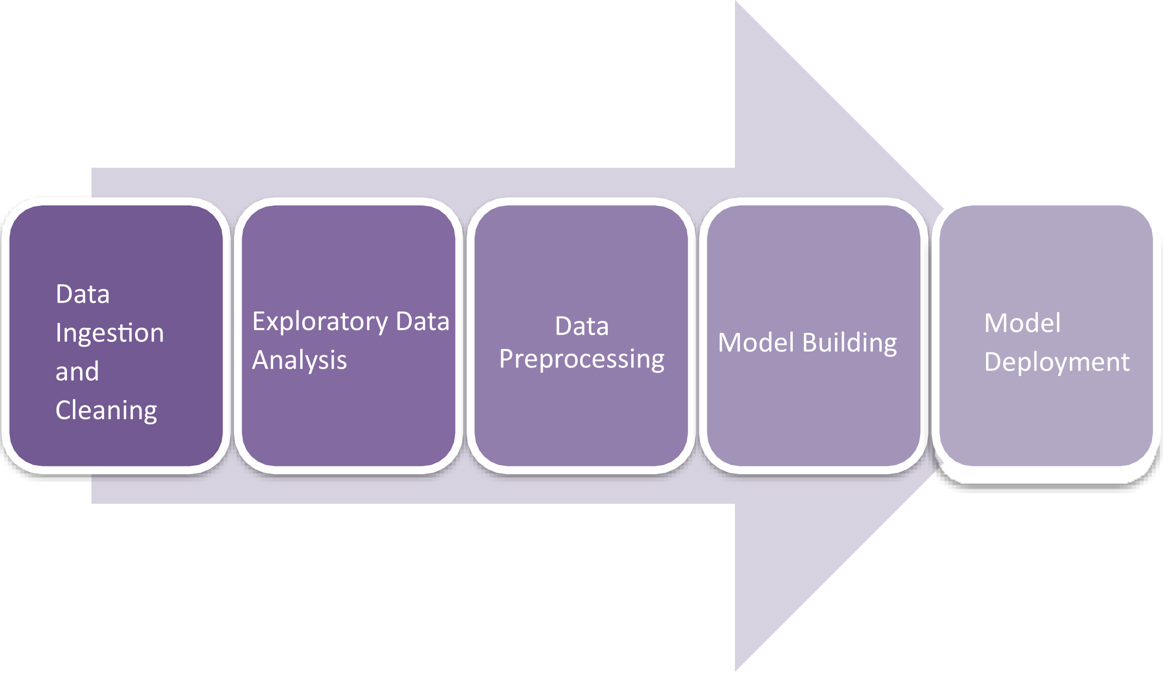
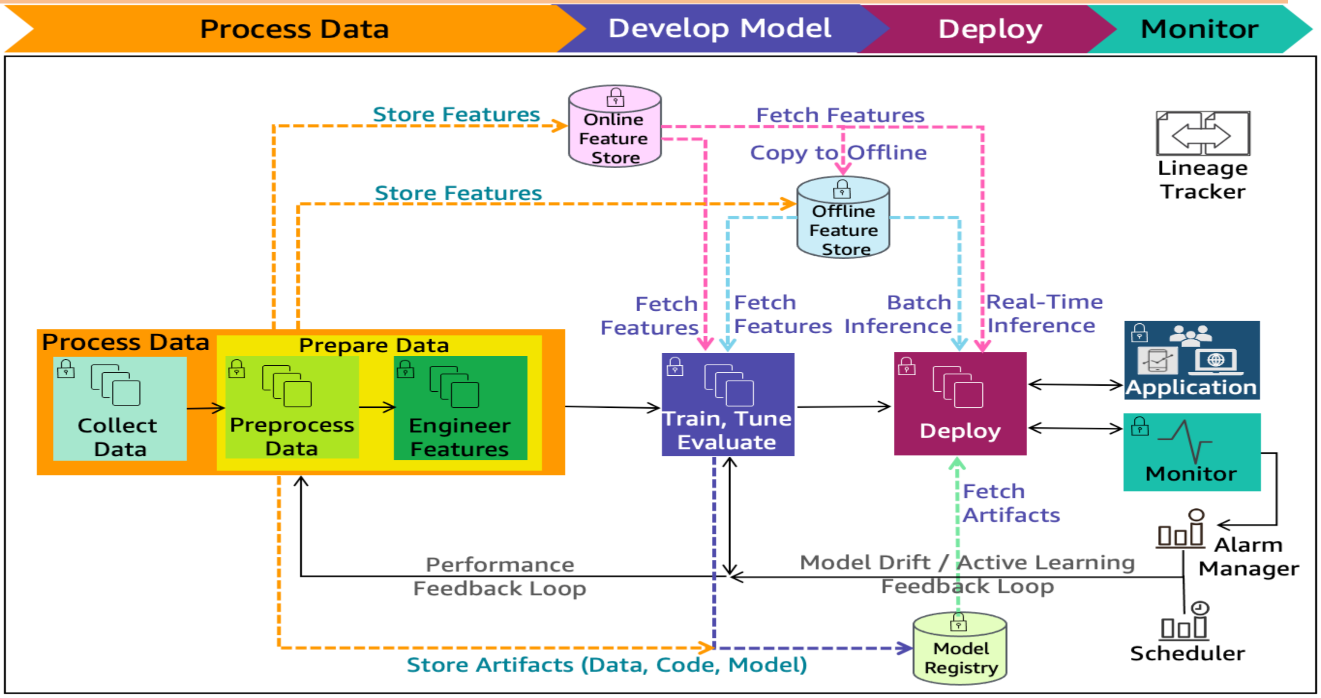


Figure 1: Functional Architecture



3.2 ML Architecture here

- **Data Ingestion:** Data is loaded from a CSV file using pandas.

- **Data Cleaning:**

  - Remove unnecessary columns and handle missing values.
  - Rename columns for clarity.

- **Exploratory Data Analysis (EDA):**

  - Convert categorical data to numerical using LabelEncoder.
  - Generate a correlation matrix and heatmap for feature selection.

- **Preprocessing:**

  - Split data into features (X) and target variable (y).
  - Apply KNNImputer for missing value imputation.
  - Use PCA for dimensionality reduction and MinMaxScaler for normalization.

- **Model Building:**

  - Split data into training and testing sets.
  - Train multiple classifiers (DecisionTree, RandomForest, KNN, SVM, Logistic Regression).
  - Evaluate models using accuracy scores.

- **Model Deployment:** The best performing model can be deployed as a prediction tool for thyroid disease diagnosis.

Explanation of this architecture

This machine learning architecture is designed to predict thyroid disease using clinical and laboratory data. It starts by loading the dataset from a CSV file using pandas for data manipulation and analysis. The data is cleaned by removing unnecessary columns and handling missing values, ensuring the dataset's quality and integrity. Exploratory Data Analysis (EDA) is performed to understand the dataset better, converting categorical data to numerical format using LabelEncoder, and generating a correlation matrix and heatmap for feature selection.

In the preprocessing stage, the dataset is split into features (X) and the target variable (y). Missing values are imputed using the KNNImputer, and dimensionality reduction is applied using Principal Component Analysis (PCA) to reduce the number of features while retaining important information. Features are then scaled to a common range using MinMaxScaler.

For model building, the dataset is split into training and testing sets. Several machine learning algorithms such as DecisionTree, RandomForest, KNN, SVM, and Logistic Regression are trained on the training set and evaluated using accuracy scores on the test set. The best performing model is selected for predicting thyroid disease.

Once the model is trained and evaluated, it can be deployed as a prediction tool for thyroid disease diagnosis. This tool provides healthcare professionals with an efficient and accurate method for early detection and management of thyroid disorders, ultimately improving patient outcomes and quality of care. The architecture provides a robust framework for developing and deploying machine learning models in healthcare settings, demonstrating the potential of machine learning in improving healthcare outcomes.

4. Model

4.1 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) plays a crucial role in understanding the dataset and preparing it for model building. Here's how EDA is used in this contextTotal Revenue

1. **Handling Categorical Data**
2. **Correlation Analysis**
3. **Data Visualization**
4. **Feature Selection**

4.2 Preprocessing and Model Building

Preprocessing: The dataset is first cleaned by removing unnecessary columns and handling missing values using the KNNImputer. Categorical data is converted into numerical format using

LabelEncoder. Dimensionality reduction is applied using Principal Component Analysis (PCA) to reduce the number of features while retaining important information. Features are scaled to a common range using MinMaxScaler to ensure each feature contributes equally to the model.

Model Building: The preprocessed dataset is split into training and testing sets. Several machine learning algorithms, including DecisionTree, RandomForest, KNN, SVM, and Logistic Regression, are trained on the training set. Each algorithm learns from the data to predict whether a patient has thyroid disease. The models are evaluated using accuracy scores on the test set, with the model achieving the highest accuracy being selected as the final model for predicting thyroid disease.

5. Model Evaluation

Model evaluation for predicting thyroid disease involves training machine learning models on clinical and laboratory data and evaluating their performance. The dataset is split into training and testing sets, with the training set used to train the model and the testing set used to evaluate its performance. The models make predictions on the test set, and these predictions are compared with the actual values to calculate metrics such as accuracy, precision, recall, and F1-score. These metrics provide insights into the model's ability to correctly classify cases of thyroid disease. To ensure the reliability of the evaluation, cross-validation techniques like k-fold cross-validation can be used. This involves splitting the dataset into k folds and training the model on k-1 folds while evaluating it on the remaining fold. The process is repeated k times, and the average performance across all folds is calculated to provide a more robust evaluation of the model's performance.