Software Requirements Specification (SRS)

Naive Bayes Classifier

1. Introduction

The goal of the Naive Bayes Classifier project is to develop a system that predicts the presence or absence of diabetes in patients based on health-related attributes. This document outlines the necessary requirements for the successful implementation of the system.

2. System Overview

The system is composed of several key components that work together to achieve the desired outcome:

- Data loading and preprocessing module: This module handles the loading of the dataset from a CSV file and performs necessary preprocessing steps such as handling missing values and preparing the data for analysis.

- Training and testing module: This module takes the preprocessed dataset and splits it into training and testing subsets. It then trains the Naive Bayes classifier using the training data and generates predictions for the testing data.

- Evaluation module: The evaluation module measures the performance of the Naive Bayes classifier by comparing the predicted labels with the actual labels using an accuracy score. This provides an indication of how well the system is able to predict the presence or absence of diabetes.

- Visualization module: The visualization module offers visual representations of the classification results, aiding in the interpretation and understanding of the predictions made by the system.

3. Functional Requirements

3.1 Data Loading and Preprocessing Module

- Load the dataset from a CSV file, ensuring compatibility with the system.

- Handle any missing values in the dataset appropriately to ensure accurate analysis.

- Perform necessary data preprocessing steps, such as scaling or encoding categorical variables, to prepare the data for analysis.

3.2 Training and Testing Module

- Split the dataset into separate training and testing subsets, allowing for the training of the Naive Bayes classifier on the training data and subsequent prediction on the testing data.

- Implement the Gaussian Naive Bayes algorithm, which is well-suited for handling both categorical and continuous features in the dataset.

3.3 Evaluation Module

- Compare the predicted labels generated by the Naive Bayes classifier with the actual labels from the testing data using an accuracy score.

- The accuracy score provides an evaluation of the system's performance in accurately predicting the presence or absence of diabetes.

3.4 Visualization Module

- Generate visualizations, such as charts or graphs, to visually represent the classification results obtained from the Naive Bayes classifier.

- These visualizations aid in the interpretation and understanding of the predictions made by the system, providing valuable insights.

4. Non-functional Requirements

The following non-functional requirements should be considered during the development of the system:

- Performance: The system should process the dataset efficiently, ensuring timely predictions.

- Usability: The system should be user-friendly, providing clear instructions for loading the dataset, initiating the analysis, and interpreting the results.

- Reliability: The system should handle exceptions gracefully, providing appropriate error handling and messages when encountering issues.

- Compatibility: The system should be implemented using Python, utilizing relevant libraries for data handling, preprocessing, training, testing, and visualization.

5. Constraints

There are specific constraints that should be taken into account:

- The dataset used must be in CSV format, ensuring compatibility with the system's data loading and preprocessing module.

- The dataset should contain relevant attributes that have a significant impact on predicting the presence or absence of diabetes, ensuring the accuracy and relevance of the analysis performed by the system.