Software Requirements Specification

Agile Software Engineering and DevOps

Prepared by:

Srihari R Acharya

Contact: sriharira.btech22@rvu.edu.in

Instructor:

CVSN Reddy

Course:

Agile Software Engineering and DevOps

Teaching Assistant:

Divya

Date:

06/10/2023

Table of Contents

Contents

List of Revisions

1. Introduction

1.1 Purpose of this Document

This document serves as the Software Requirements Specification (SRS) for the "Prediction of Heart Attack Using Machine Learning Algorithms and Deep Learning" project.

1.2 Project Scope

The project's objective is to implement and optimize a range of machine learning algorithms and deep learning techniques to predict heart attacks. The project encompasses the following components:

- Implementation of classification algorithms (such as Decision Tree, Random Forest, SVM, Logistic Regression, KNN, and Naïve Bayes) with customizable parameters for enhanced accuracy.

- Development of a Deep Neural Network (DNN) for heart attack prediction, with parameters like the number of neurons per layer, epochs, hidden layers, and activation functions that can be adjusted.

1.3 Target Audience and Document Overview

This document is intended for the project team, including developers, testers, project managers, and the course instructor. It offers an overview of the project's requirements and constraints.

1.4 Definitions, Acronyms, and Abbreviations

- DNN: Deep Neural Network

- SVM: Support Vector Machine

- KNN: K-Nearest Neighbours

- SRS: Software Requirements Specification

1.5 Document Formatting

This document adheres to the IEEE standard format, employing Arial font size 11, single-spacing, and 1" margins. It consistently employs the IEEE section and subsection titles template.

1.6 References and Acknowledgments

This document does not require external references.

2. Project Overview

2.1 Product Description

The "Prediction of Heart Attack Using Machine Learning Algorithms and Deep Learning" project involves the creation of a software application that forecasts the likelihood of a heart attack using a variety of machine learning and deep learning algorithms. The application will rely on the Heart Attack dataset for both training and testing.

2.2 System Functions

The principal functions of the system comprise:

- Implementing various classification algorithms for heart attack prediction.

- Configuring algorithm parameters to optimize accuracy.

- Incorporating a Deep Neural Network for heart attack prediction.

- Enabling users to fine-tune DNN parameters, including the number of neurons in each layer, epochs, hidden layers, and activation functions.

2.3 Design and Implementation Constraints

- The project must strictly adhere to Agile software engineering and DevOps practices.

- Python and relevant libraries for machine learning and deep learning will be used for the implementation.

- The availability of the Heart Attack dataset for training and testing is assumed.

2.4 Assumptions and Dependencies

- The project operates under the assumption of access to the Heart Attack dataset for training and testing.

- The project is dependent on the availability of Python and requisite libraries.

3. Detailed Requirements

3.1 External Interfaces

3.1.1 User Interfaces

The user interface will be designed as a command-line interface (CLI) for simplicity.

3.1.2 Hardware Interfaces

The system does not necessitate specific hardware interfaces.

3.1.3 Software Interfaces

The project may need to interface with external libraries relevant to machine learning and deep learning.

3.2 Functional Requirements

- The system will implement the following classification algorithms: Decision Tree, Random Forest, SVM, Logistic Regression, KNN, and Naïve Bayes.

- Users will have the ability to define training and test data sizes for algorithm fine-tuning.

- The system will feature a Deep Neural Network (DNN) for heart attack prediction.

- Users will have the flexibility to adjust DNN parameters, including the number of neurons in each layer, epochs, hidden layers, and activation functions.

3.3 Use Case Model

[Include a use case diagram and specifications for each use case.]

3.4 Non-functional Requirements

3.4.1 Performance Requirements

- The system is expected to deliver results with high accuracy and minimal latency.

- It should be capable of efficiently handling substantial datasets.

3.4.2 Safety and Security Requirements

- The system should include rudimentary security measures for data and user interactions.

- The privacy and confidentiality of user data must be assured.

3.4.3 Software Quality Attributes

- The software should be designed for maintainability and adaptability to various datasets.

- It must prioritize accuracy in prediction.

- The system should offer user-friendliness and ease of use.

Appendix A – Data Resources

Heart.csv

Appendix B - Project Log