Capstone Project Proposal

Nadima Dwihusna

January 4, 2021

I. Domain Background

The Titanic was widely considered as the “unsinkable” ship. However, during its voyage on April 15, 1912, the Titanic sank after hitting an iceberg. Because there were not enough lifeboats, 1502 out of 2224 passengers died. Using the passenger data (e.g. name, age, gender, socio-economic class, etc.) that boarded the Titanic during the voyage, there seems to be a pattern where some groups of people were more likely to survive than others. The main objective of the project is to use machine learning to create a model that predicts which passengers survived the Titanic shipwreck.

This project is taken from the Kaggle Competition: “Titanic – Machine Learning from Disaster” (https://www.kaggle.com/c/titanic/overview/description). I would like to tackle this project as this will be my first Kaggle competition and first time building a predictive machine learning model.

II. Problem Statement

As stated in the Kaggle Competition (https://www.kaggle.com/c/titanic/overview), the main goal of this project is to build a machine learning model that answers the following question: “What sorts of people were more likely to survive?”. The data of the passengers who boarded the Titanic in 1912 is provided and to be used to build the model that predicts if the passenger survives.

III. Datasets and Inputs

The training and testing data set can be found in the Kaggle Competition website (https://www.kaggle.com/c/titanic/data). The data is split into two groups:

- **training set (train.csv)**
- **test set (test.csv)**

The training set will be used to build the machine learning models. This training set contains the outcomes (ground truth) for each passenger. The test set will be used to see how well the model performs with new data. The test set does not have ground truth for each passenger as it is my job to predict the outcomes (passenger survive or not). Additionally, a **gender_submission.csv**

file is provided. This file shows an example of what a submission file should look like. All the features in the dataset describes the passenger information as follow:

Feature Variable	Definition	Key
PassengerId (integer)	Passenger ID / count number	
Survived (integer)	Survival	0 = No, 1 = Yes
Pclass (integer)	Ticket class (socio-economic status or SES)	1 = 1st, 2 = 2nd, 3 = 3rd
Name (string)	Passenger name	
Sex (string)	Sex	
Age (integer)	Age in years (Age is fractional if <1. If the age is estimated, it is in the form of xx.5)	
SibSp (integer)	# of siblings / spouses aboard the Titanic	
Parch (integer)	# of parents / children aboard the Titanic	
Ticket (integer)	Ticket number	
Fare (float)	Passenger fare	
Cabin (string)	Cabin number	
Embarked (string)	Port of Embarkation	C = Cherbourg, Q = Queenstown, S = Southampton

IV. Solution Statement

The solution is a machine learning model that predicts if the passenger would survive based on the given passenger information / features. An endpoint will be made using Python Frameworks to have predictions through HTTP. Python libraries such as Pandas and NumPy will be used for data processing. The model will be built using the scikit-learn estimators and the quality will be measured using the scikit-learn performance metrics or loss function. In the end, Python Flask and Docker will be used to make an endpoint to access the model for predictions.

V. Benchmark Model

A benchmark model will first be made using the scikit-learn estimators with the default “standard” hyperparameters that relates to the domain, problem statement, and solution. Afterwards, the

model will be optimized with different hyperparameters. The benchmark model will then be used to objectively compare and evaluate the results with the alternative models. This will help identify improve the final model performance.

VI. Evaluation Metrics

An accuracy score and confusion matrix of the classification results will be used as an evaluation metric to quantify the performance of both the benchmark model and the solution model.

VII. Project Design

1. **Visualization, Feature Engineering / Extraction, and Data Preprocessing**
   The first step is to visualize and get a better understanding of the data and its features. Additionally, the Principal Component Analysis can be used to identify the feature importance and select the most meaningful features to be used for training the model. As stated previously, train.csv represents the training data and test.csv represents the testing data. In this step, for both the training and testing dataset, any outliers or empty data should be removed to clean the data, and all the features should be normalized. For the train.csv training data, the data would then be split 85% for training and 15% for validation. These training data sets should be balanced to provide enough training data for each target class. The test.csv testing data will be used for the end model evaluation.

2. **Algorithm Selection**
   In this step, different kind of scikit-learn estimators with default standard hyper paraments will be trained. To start, I plan to train using simple models such as K-Nearest Neighbors, Support Vector Machine, and Random Forest. The best model will be selected, and the initial benchmark model will be created.

3. **Train Model**
   The model hyperparameters will be tuned and optimized in this step using grid search for comparison with the initial benchmark model.

4. **Testing and Evaluate Results**
   The model will be tested to predict which passengers survive the shipwreck. The scikit-learn performance metrics or loss function will be used in this step.