CHAPTER 5.0

SYSTEM DESIGN

5.1 DESIGN PHASES

The whole project is divided into 3 phases.

- Data Preprocessing
- Model Generation and its testing
- GUI Development for better user interaction.

5.1.1 Data Pre-processing

This stage involves fetching data from web and then processing it to make it suitable for modelling. The data I obtained is in the form of csv files where each file describes each match details. The details that single csv file constitutes are match id, team names, toss winner, toss decision taken after winning, venue of the match, date of the match, scores of each team after every over. This data also contains few missing entries. With the help of python, I first parsed all the files and stored the data in a single file. While parsing I have fetched only required attributes from each file rather than storing whole contents of file. Now the resultant file each row corresponds to details of each match. After parsing I handled missing values by taking mean of that particular feature for which the value is missing.

5.1.2 Model Generation and its testing

After pre-processing I used the machine learning to generate models which can be used to predict the results. I used different machine learning models and calculated the accuracy from each model and finally selected the one with best accuracy. I used decision tree, random forest, logistic regression and SVM. Out of all these I used logistic regression as it gave us the better accuracy among all other models. While generating the models I used K Fold cross validation technique so as to make use of whole dataset as training as well as testing. For future match prediction I first take the inputs from user via GUI which is created using python flask framework. Then I first convert those inputs in numeric form like 1 for team A and 0 for team B and generate other important features using the modules I have

used in phase 2. Then we convert these whole inputs into panda's data frame and pass it to model for prediction. The model does the processing and gives us the winner of the match.

5.1.3 GUI Development for better user interaction

In this phase I developed a simple user interface so that a user need not to go to command prompt for entering all the required inputs. The interface has been developed using python framework flask. The interface contains a single form where user can enter some match details. The form developed also validates if the provided input is valid or not. If the entered information is invalid it returns an error message. Also, if the user doesn't give all the required inputs then it displays an error message to fill all the required entries. Thus, it is easy and better way of getting all the valid inputs.

5.2 USER INTERFACE SNAPSHOTS

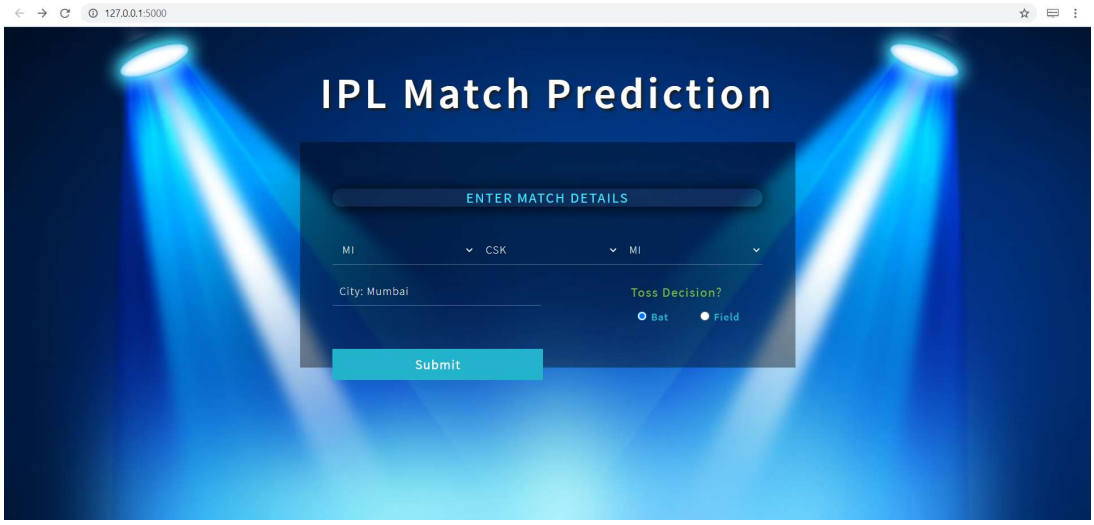

Figure 5.1 Home page

Figure 5.2 Pridected Output Page