Software Design Project 3: Text Mining and Analysis

Taejin Kim

26 February 2015

1 Project Overview

This project analyzes the correlation between top soccer players' popularity and their ranking. Many have questioned whether the World Player of the Year (Ballon d'or) award has been unfair at recent times. The project will pull text from Google search, and keep track of word frequencies of players names in the search results.

2 Implementation

2.1 Process Overview

Initially, the code runs multiple Google searches and stores all the resulting text from relavent web pages in a single text file. The Google searches are predetermined by the user, and in this case includes generic football related terms like 'soccer','European Football','World Cup', and so forth.

The code then processes and analyzes the mined text. The code removes all punctuation in the text and converts all letters in the text to lowercase. With the altered text, a histogram in the form of a dictionary is created. The words of the text are the dictionaries keys, and the frequency of the word in the text is the value. Finally, a new dictionary based upon the old one is created, where the keys are the predetermined player names, and the values are the values are the frequencies of the player's name in the processed text.

2.2 Decisions

When developing the code, the code could have either taken the processed text and created a list with each word split as an element in the list, and could have iterated through that list and created a dictionary with just the player names as the keys in a histogram style way. However, I decided to use Allen Downey's functions that takes a block of text and creates a histogram-dictionary with every word that appears in the text. I decided to take the 'leap of faith' by using Allen Downey's code, that I know works. Moreover, pulling keys (player names) out of a histogram seemed easier than iterating through a very long list of sliced text, as it simplifies the final function that I call from the terminal, making debugging easier. Other small decisions include creating a new empty text file every time the code runs to make the code runnable from any directory, and using lower case single strand letter for player names so it is easier to correlate to the data earned from the histograms.

3 Results

After counting the number of player names from the generic soccer-related searches, the following shows the players' name to the frequency they appeared:

{'toure': 1, 'messi': 97, 'kroos': 0, 'iniesta': 4, 'gotze': 1, 'neymar': 14, 'courtois': 2, 'ibrahimovic': 6, 'rodriguez': 7, 'robben': 5, 'mascherano': 6, 'bale': 5, 'schweinsteiger': 0, 'hazard': 9, 'neuer': 28, 'ramos': 2, 'muller': 1, 'maria': 1, 'pogba': 0, 'costa': 10, 'ronaldo': 121, 'benzema': 2, 'lahm': 2}

When we compare the frequency by Google search count to the number of votes the top 10 players got for the Ballon d'or award, there is a striking similarity between the proportion of votes to the proportion of the frequency of the player names in Google search.

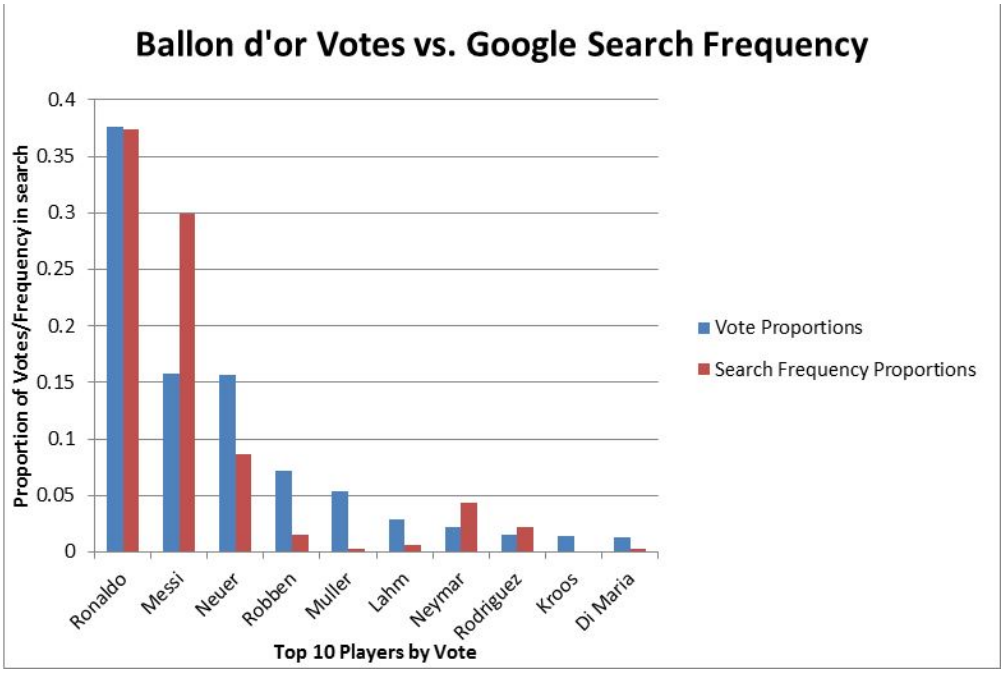

Figure 1: Vote data from goal.com

Although the frequency trend does not strictly follow that of the votes, the top three players of the votes had the top three players by frequency in the same order. Although player popularity and skill are not independent from each other, there is a definite trend between how many times a player name appears in Google, and the number of votes they get. Therefore, the evidence suggests that the Ballon d'or may be influenced by how popular a player is or how much publicity they receive.

4 Reflection

For this project, the non-data mining portion of the code went very well. Creating a histogram and pulling the wanted key-value relationships were very simple. Because I am implementing Allen Downey's code in creating histograms, I took the 'leap of faith' and did minimal unit testing. Unit testing for the data mining was done by opening the text file and looking at all the google results present in the file. However, I struggled with mining data from google. Trouble with the license key, and converting html to txt files took many hours of debugging with the help of a ninja. The scope of my project may have been a little small, as all I did was a simple frequency analysis, but the mining of the data was the major portion of the project. If I knew about the trouble with google beforehand, I may have taken lots of text from a simpler source like project Gutenburg, and did a more sophisticated analysis on that data. I did get help using a dictionary, and updating it, along with reading, writing, and using a separate text file, which are parts of python I was not completely comfortable with.