COMP20008 - Assignment 1

Lucas Fern (1080613)

April 11, 2020

Web Crawling and Scraping of Tennis Data

The aim of this project was to demonstrate an understanding of basic web scraping, crawling and data processing skills by extracting, refining and presenting tennis data stored in a collection of news articles on a web server. This project was broken down into 5 individual tasks which will now be discussed in detail.

Task 1 - Crawling

Task one was the web crawling component of this project. My method involved:

1. Extracting any links present on the webpage at the root URL, (`http://comp20008-jh.eng.unimelb.edu.au:9889/main/`) and adding them to a python set. All the webpage data for the entirety of the project was accessed using the python `requests` module and processed with `BeautifulSoup`.
2. Adding any visited URLs to a different set, then - while the set of visited URLs did *not* contain the entire set of URLs to visit - visiting an unvisited link and scraping its heading as well as any outgoing links.
3. For each link visited above, the URL and the heading extracted from the `<h1>` HTML tag was stored in a `DataFrame` provided by the python module `pandas`.
4. Finally, after the set of visited links included the entire set of links to visit, the URL and heading data was output to a CSV file using the `pandas.DataFrame.to_csv` method.

The output from this task was a 101 line `.csv` file with a heading row containing headings `url` and `headline`, each followed by 100 entries of those data from the 100 scraped articles.

Task 2 - Scraping

Task 2 involved finding the first player that was mentioned in each article that we found in task 1, and extracting a valid tennis game score from it. Both of these pattern matching steps were accomplished with the use of regular expressions provided through the python `re` module.

To match player names, the valid player names first had to be extracted from the provided `tennis.json` file. This was accomplished by opening the file using `json.load()` from python's `json` module and extracting the name value from each player's entry. This list of `n` names was then converted to lowercase and then compiled into a regular expression in the format

```
re.compile('((firstname_1 lastname_1)|...|(firstname_n lastname_n))')
```

where the `...` are replaced with the other names in the list enclosed in parenthesis.

After extracting all of the text present in each page's heading and body tags and converting it to lowercase, we can then use this regular expression to identify and store the first player mentioned in the article.

Next, extracting the first valid tennis score from each page involved a few steps. Firstly the following regular expression was compiled:

```
re.compile(r'(((\d+-\d+)|(\(\d+[-/]\d+\)))[. ,]){2,}')
```

This expression matches any set of two or more groups of hyphen separated integers (or tiebreaker sets separated by '`/`'). These groups can be separated by spaces, but also commas or full stops, as in testing it was discovered that some articles contained scores punctuated in this way. We also match set scores enclosed in parenthesis by escaping them in the regular expression such as `\)` since this is valid notation for a tiebreaker set.

Unfortunately, this regular expression matches some invalid game scores, such as when neither player has reached

6 points. Because of this, python code was added to ensure that the game score extracted was complete by checking that either 2 or 3 games had been won by a player and that each set in these games reached a score of at least 6, or more in the case of a tiebreak. In the case an invalid score is found, the regular expression is used in an iterator to return the next score in the webpage until a valid score is found.

Once the player data and scores have been extracted and validated, they are added to a `DataFrame` only if both items are present for a given URL. Finally this is exported to a `.csv` file, containing 4 columns, `url, headline, player` and `score`.

My program identified 41 articles that contained both valid players and scores, though I believe this would be significantly higher if the `tennis.json` file contained data for female players as well as males. This resulted in a CSV file with 41 rows of data, where set scores were separated by spaces. An example row looked like:

```
http://comp.../enmanove001.html,Henman overcomes rival Rusedski,Tim Henman,4-6 7-6 (8-6) 6-4
```

Tasks 3-5 - Data Processing, Presentation and Analysis

Tasks 4 and 5 involved displaying a selection of the data extracted in previous tasks. Plots for both of these tasks were generated with the python module `matplotlib.pyplot`. In task 4 the data from task 2 was used to sum up the amount of articles written about each player, this was done with the `pandas.DataFrame.groupby()` method. The top 5 players who were the subject of the most articles were then extracted and this data was displayed on the boxplot in figure 1.

This clearly shows that the top two players were the subject of 5 articles each and players 3-5 the subject of 4.

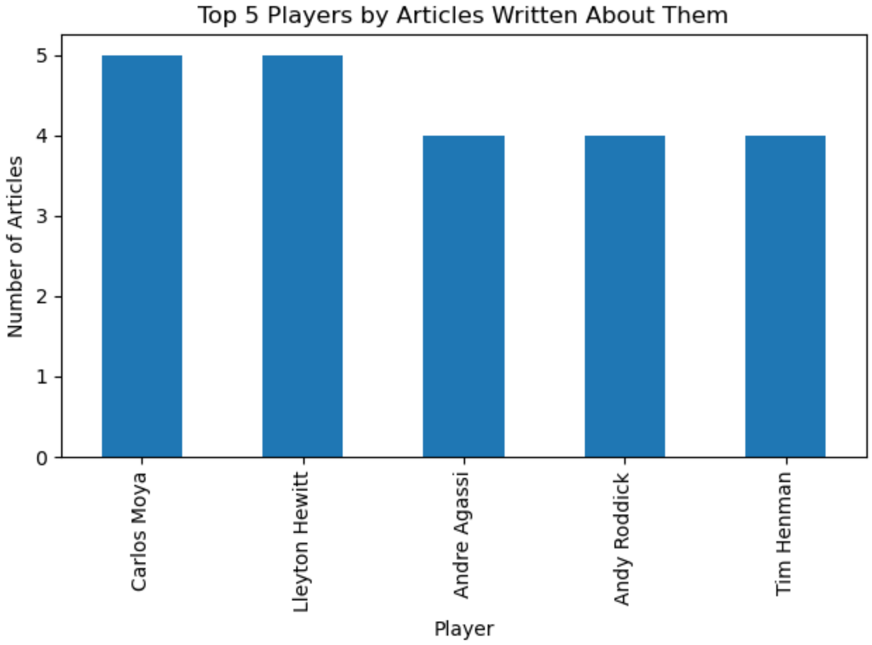

Figure 1: Bar chart of the amount of articles written about the tennis players who are the subject of the most articles.

It will later be discussed, however, that it was potentially inappropriate and inaccurate to associate each article to only the first player mentioned, thus in reality the actual number of articles these players feature in may be substantially greater than what is shown here.

Figure 2, generated in task 5 uses a new dataset that was generated during task 3. In task 3 the *game difference* was extracted for each article identified in task 2. This was done by finding the difference between each players cumulative set scores in each article, then the average game difference was found by taking the mean of each players game differences.

Figure 2 shows each player's average game difference plotted in a double bar chart with their win percentage, which was taken from `tennis.json`. By sorting this figure by decreasing win percentage it can be seen that there is no clear correlation between their win percentage and the average game difference extracted here. This is likely because of a few factors, including the small sample size that resulted in many players only registering a score for a single game, as well as the fact each article's first score was associated with the first player

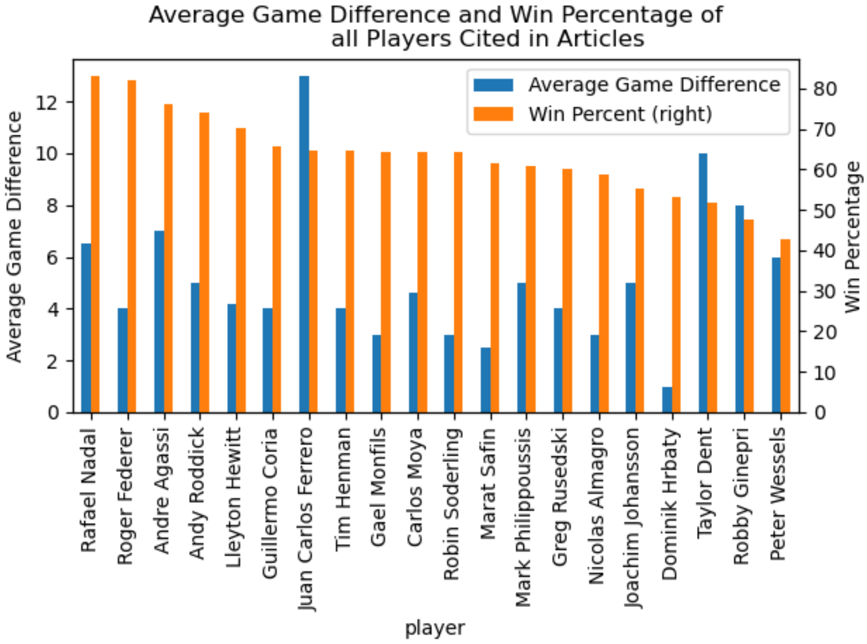

Figure 2: Double bar chart of the win percentage and average game difference of each player mentioned in at least one article. Sorted by decreasing win percentage.

mentioned. In a larger more accurate dataset it would be reasonable to expect a correlation between increasing win percentage and increasing game difference as more skilful players would be expected to both win more games and win sets by a larger margin.

The fact that the game difference was taken as an absolute value was mandated by the fact that it was not identified whether or not the player won or lost their game. This demonstrates another flaw in the methodology, as a player who was the subject of many articles for suffering numerous crushing defeats would appear to have an unreasonably large game difference. It is perhaps because of this that there is little correlation between the a players win rate and their average game difference.

Discussion

Appropriateness of Associating Names to Scores

In task 2 each article was scraped for the first player name and valid game score present. These values were stored together and the assumption was made that the score found related to only that player. There are a variety of reasons why this is a far from perfect assumption. Firstly each game is played by two players, by associating the score to the first mentioned player we are not only assuming that they played in that game, but are neglecting their opponent. To remedy this partially, it may be more appropriate to try and identify a second player mentioned in the article and attempt to resolve a winner and loser of the game. A method for this is discussed in the next section.

Since the vast majority of the scores in this dataset were ordered such that the winning players scores appeared first in the set score, it is somewhat reasonable to think that the winning player would be mentioned first in a majority of articles. This is however, not true in all cases such as in the article “Safin slumps to shock Dubai loss,” and as a result of not being able to determine the winning player in this way, the game difference had to be taken as an absolute value in task 3, meaning it did not correlate closely with a player’s skill.

Overall this assumption served to significantly reduce the amount of processing that had to be done on the dataset, but resulted in the introduction of a range of errors, thus it is not a very appropriate assumption.

Suggestion for Identifying Winning Players

It would be possible to infer the victor of many of the articles in the dataset by using a tool such as regular expressions to match for certain words nearby to a players name in an article. For example, in multiple articles the losers name is followed closely by the words ‘loss’ or ‘defeat’ (eg. ‘**Andre Agassi** suffered a comprehensive **defeat**...’, ‘**Marat Safin** suffered a shock **loss**...’) and conversely, words related to victory, as in ‘...**Roger**

Federer won...' may be used to identify a player as victorious.

This is however, not a perfect system, as, without comprehensive checks, phrases like '**Tim Henman** saved a match point before fighting back to **defeat** British rival Greg Rusedski' may falsely identify Henman as the loser of this game. Because of this, more advanced language processing techniques may be required to achieve greater accuracy.

Other Data That Could be Extracted

To gain further insight into player performance, more analysis could be done on the text contents of each article. Studying the vocabulary used in articles about each player and extracting a selection of words to concisely summarise the theme of each article could be done with the use of a lemmatiser. This could then presented visually to understand how the player is generally talked about in the media, from which conclusions could be drawn about their performance with large enough datasets.