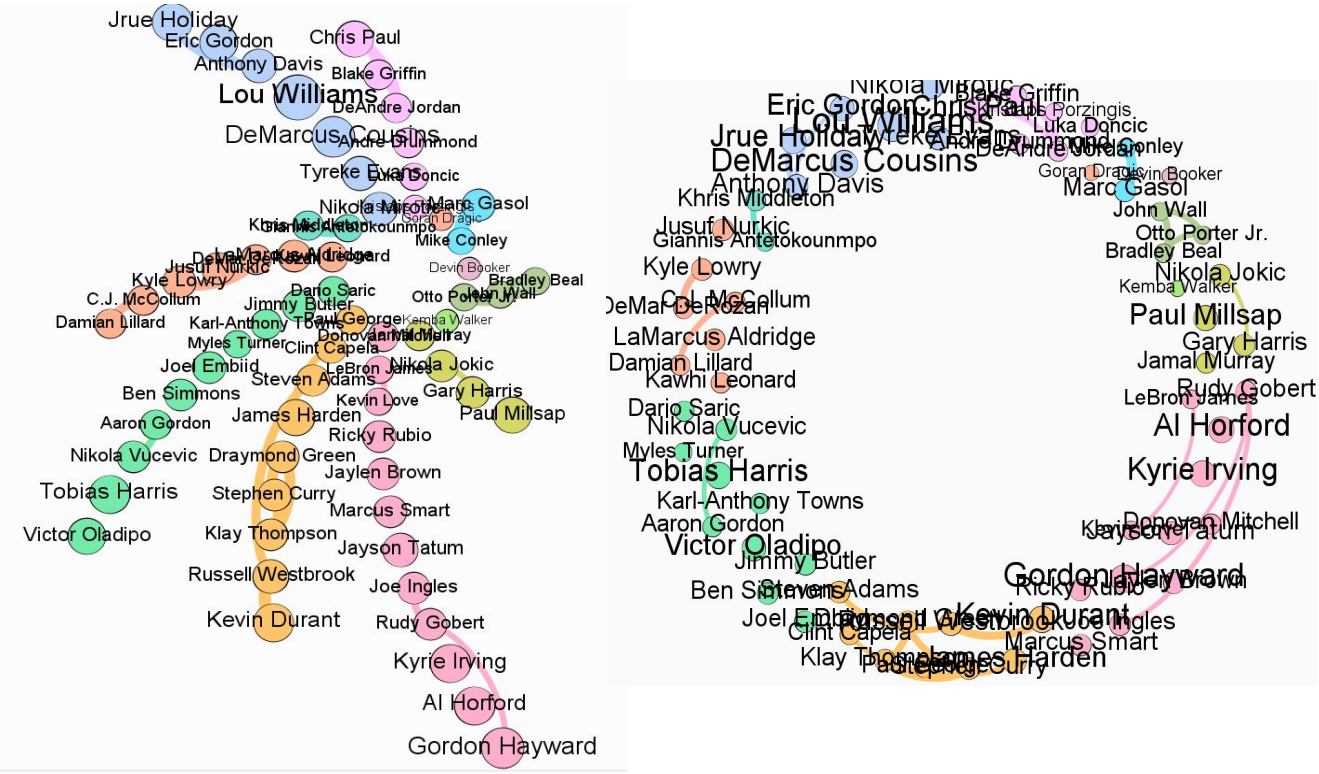
DATA SCIENCE SEMINARS

CLUSTERING NBA PLAYERS WITH GEPHI

BIG DATA VISUALIZATION



Author: José Manuel Pérez Ricote

Academic Year: 2018-2019

INTRODUCTION

The core objective of this project is to implement a big data visualization on a concrete dataset. In this case, the selected domain is basketball NBA (National Basketball Association) players. It has been chosen a list with 65 of the best players of this 2018-2019 season from a public source [1]. The only modification done to such list is the addition of Luka Doncic, the Slovenian sensation which has clearly outperformed any prediction of his skills, substituting P.J. Tucker, an average player from Houston. This list of players is implemented as nodes for a graph, with each player with a different identifier.

Moreover, the connection between nodes has been established using the number of years in which players have shared the same team. It is considered to share a team if by the end of the trade market (approximately by the 20th of February) both players are in the same team. In order to perform this arc's connection, the sources for this knowledge were taken from a specialized web page for basketball statistics, Basketball Reference [2]. None of the lists were stored in any database, so they have been created from scratch using as example a default GEXF Graph File, “LesMisérables.gexf”.

The “NBA.gexf” file is made of nodes and arcs. Both are shown in the below figures, one reveals how to create a node in a GEXF Graph File and set the configuration of the file, while the other one is in charge of creating all the connections between players that shared a team in the past or in the present.

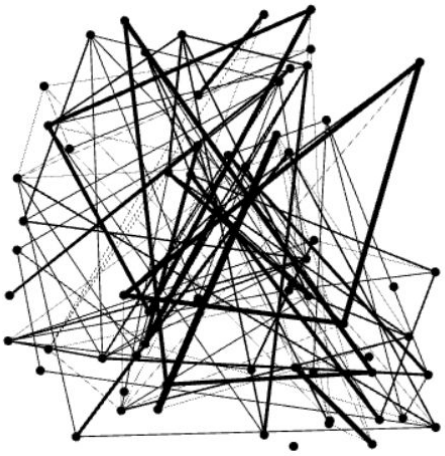
```
<?xml version="1.0" encoding="UTF-8"?>
<gexf xmlns:viz="http:///www.gexf.net/1.1draft/viz" version="1.1"
xmlns="http://www.gexf.net/1.1draft">
<meta lastmodifieddate="2019-03-23+12:44">
<creator>Jose Manuel Perez</creator>
</meta>
<graph defaultedgetype="undirected" idtype="string" type="static">
<nodes count="65">
<node id="0.0" label="LeBron James"/>
<node id="1.0" label="Kevin Durant"/>
```

```
<node id="64.0" label="Luka Doncic"/>
</nodes>
<edges count="129">
<edge id="0" source="1.0" target="4.0" weight="3.0"/>
<edge id="1" source="3.0" target="11.0" weight="2.0"/>
<edge id="2" source="6.0" target="12.0"/>
<edge id="3" source="4.0" target="13.0" weight="7.0"/>
```

Once the configuration for the big data visualization is explained, it has to be said that the objective of this task is to classify the different players in clusters of players that have shared the court as part of the same team. Moreover, it is possible to measure which players are likely to play with other outstanding players, so that they had more facilities in winning NBA titles. Otherwise, it would be possible to know which players were alone as star players in their squads.

GRAPH CUSTOMIZATION

The graph configuration starts with the import of the graph file. In this case, the graph edges are undirected, because it does not matter at all if the arcs are connected from one player to another or vice versa. It is convenient to allow loops, graph auto-scaling and to create the missing nodes. Also, the strategy to combine the different edges is by sum, instead of average, maximum, minimum, etc... Then, it is time to import it and start customizing it to reach some results. The first imported graph can be observed in the following figure:

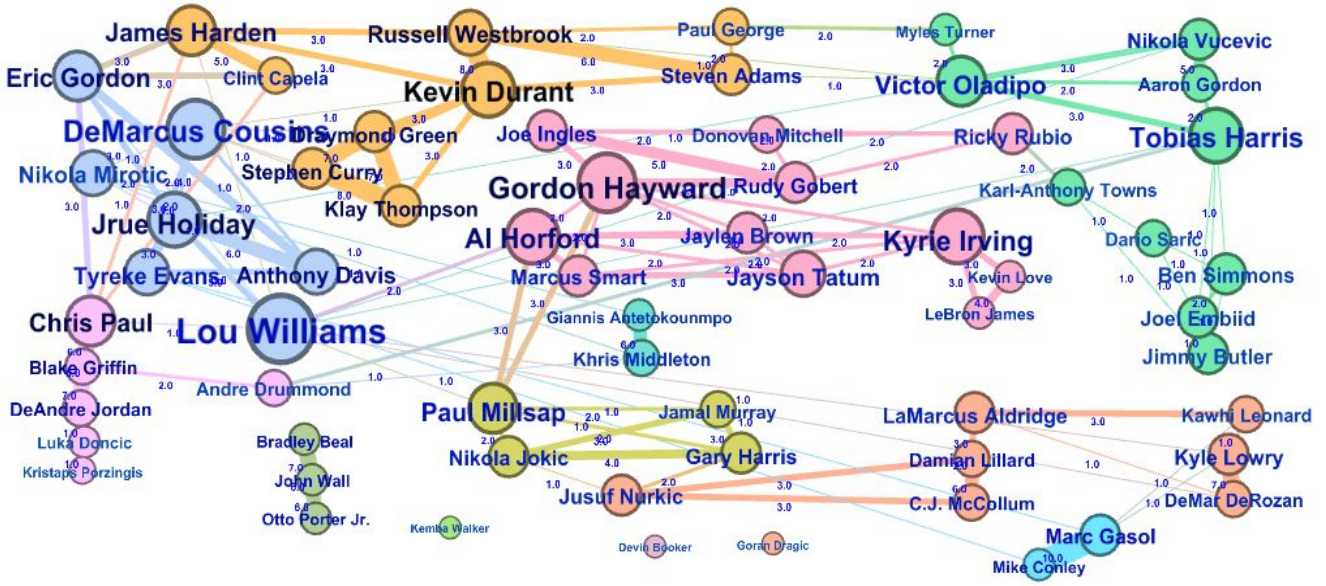


Now, we start the graph customization by computing all the

statistics that could be computed, in order to be able to rank or partition depending on such statistics. For our purpose, several strategies of this last have been used in different aspects. At first, the nodes have been clustered in different groups depending on their connections. This can be performed setting a few configuration steps. We focus on appearance section, and select as options nodes, color palette and partition. Then, it is selected as attribute the modularity class, which strengthens the relationship between nodes that are connected between them. In this way, it is possible to cluster networks as they one presented.

Afterwards, it is defined the size of the nodes depending on the relative degree of arcs that each node has. Whoever has played with more star players of this list will be represented with a bigger node, which is Lou Williams with 10. Also, we set as values in the graph the weight of each connection between players.

Finally, the label color can also be defined in this section. It is done defining another range for a concrete attribute, degree with weights, within a blue-black color scale. This means that if a player has shared team with two big stars, three years with one and two with the other, he will have a degree with weights of five. The player that has played more years with players in that list is represented in black, Kevin Durant with 24. The final graph after this section is the shown below:



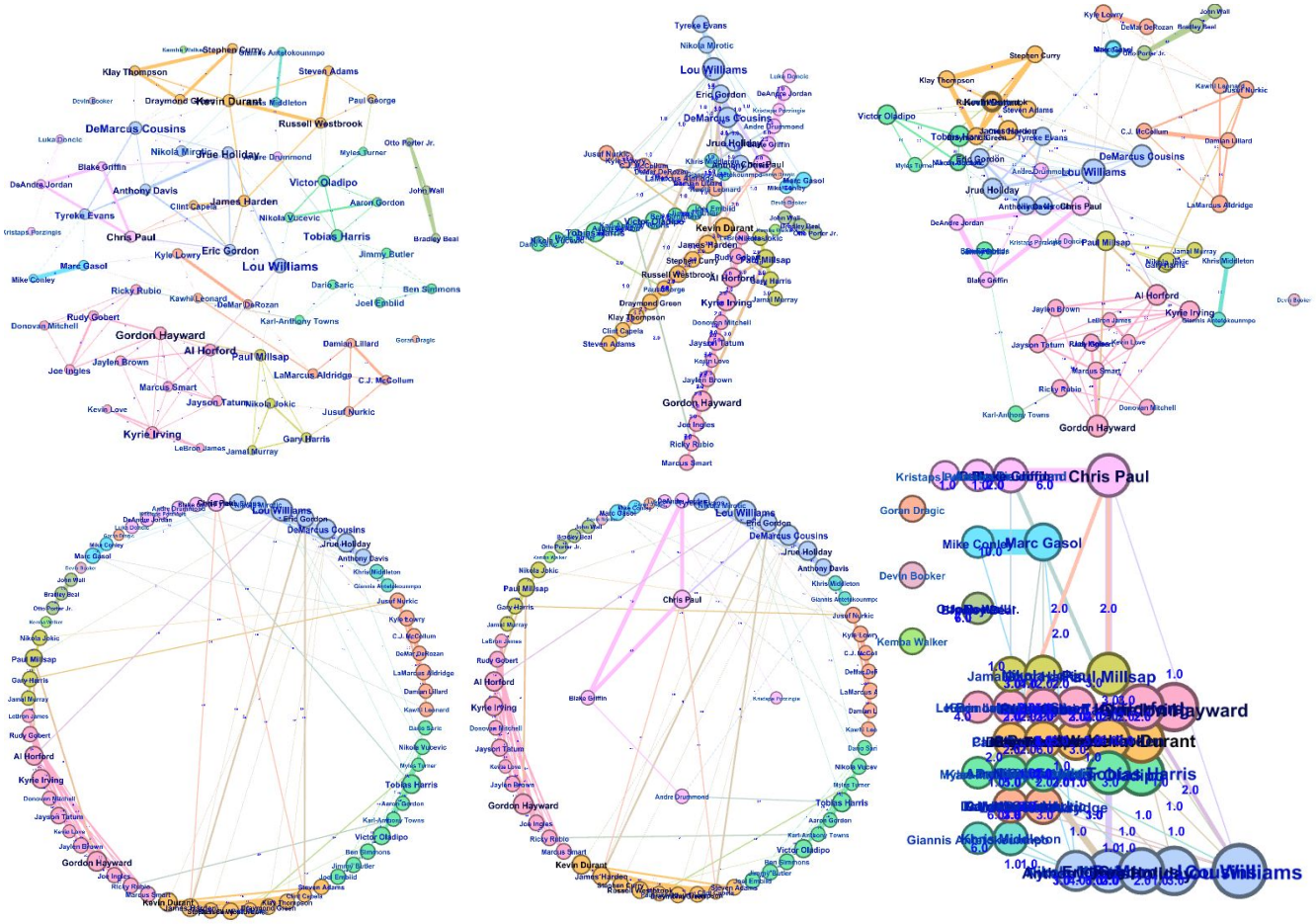
GRAPH LAYOUTS

This section is focused on applying different graph layouts to the final graph of customization part. There are lots of different layouts for Gephi graphs with different meaning and scalability towards the number of nodes and arcs. For the main objective of the project, several of them have been applied trying to reach the best visualization possible.

The first one was Fruchterman-Reingold layout, which simulates the model as if the nodes were mass particles. The second one is the radial axis layout, in which each of the axis is a concrete cluster of the modularity class. Also, the sorting in each axis is done by means of node's degree. The third approach used for visualizing the plot was OpenOrd layout, which is an extension of Fruchterman-Reingold layout that tries to address undirected weighted graphs like this one. Its objective is to create a cluster differentiation, like the one we want to perform. The fourth layout is the circular one, which shows the distribution of nodes according

to an attribute and their links. The fifth layout is pretty similar to the circular one, as it is a Dual Circle layout, in which instead of just creating a circle, there are two. Lastly, the sixth layout covered for the goal of the project was the Geo layout, which differentiates the nodes by using two attributes, one to define the latitude and another to define the longitude.

The different graphs with the layouts applied is shown in the below figures:

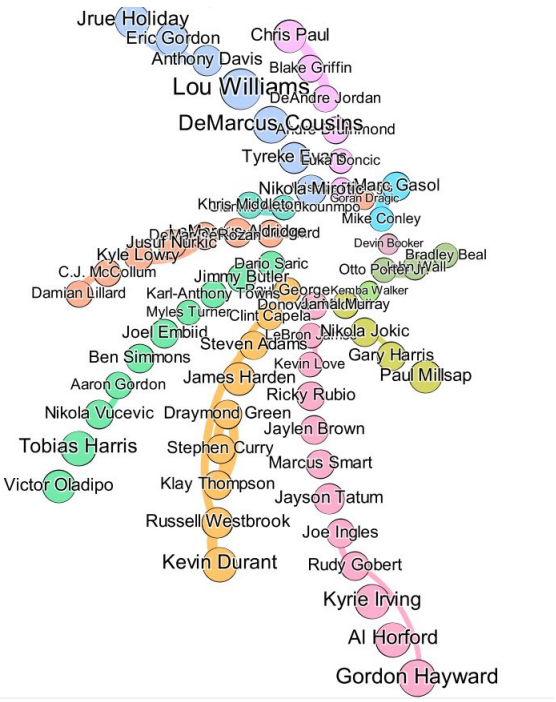


FINAL RESULTS AND CONCLUSIONS

The last section is devoted to explain the results we obtained from the whole analysis, and to define a set of important conclusions for the project. Firstly, it is important to show which is the decided layout and why. The OpenOrd layout looked like a really interesting layout for clustering the star players of the league. However, this approach is only better for graphs in which the number of nodes exceeds a hundred. When observing the resulting graph, it is not that obvious the clustering purpose of this layout. Therefore, we try to address the layouts whose objective is create a ranking for a given attribute. As we have modularity class as attribute, we may represent the ranking using the clusters as different aggregations. This is clearly performed in the radial axis layout. We set as first parameter for each axis the modularity class attribute, so each of the axis of this layout is focused on a different cluster. For the second parameter, we set as ordering for each axis the weighted degree of each node. The node that is the most separated from the center of the graph is the one with higher weighted degree. Moreover, we apply some visualization configuration to improve the

legibility and the display of the final graph. In order to do so, we should follow a set of steps.

The first of those steps is to set the spiral characteristic for the radial axis layout. Then, we create a new query for the a given parameter. In this case, it is filtering the edges whose weight is lower or equal to three. In this way, we are only plotting only the most significant connections, and deleting a lot of noise produced by tens of disposable connections. The third step would be to refresh the preview with curved edges as parameter, so that the arcs are plotted in a curved way. Finally, we refresh it again using text contour as parameter. The final graph is shown next to the explanation of the steps.



Once the final graph is obtained, it is possible to extract some results from it. At first, it can be said from the clusters that they are settled according to different teams. As an example, Kevin Durant and orange cluster corresponds to Golden State Warriors, Oklahoma City Thunder and Houston Rockets franchises. These three franchises share a common fact, as they all have one of their star players that was a former “Big Three” player of Oklahoma City Thunder 2011-2012 NBA finals team (Kevin Durant, Russell Westbrook and James Harden). Analysing other cluster, the yellow one is only made by players which are currently part of Denver Nuggets squad, which is performing really well this 2018-2019. These players have all played together for at least two years.

Therefore, with all the results analyzed, some conclusions have been extracted from the whole process. At first, it should be said that it is possible to know which are the players who had the opportunity of playing with more star players in their same team, like Lou Williams. This is because Lou is a player who has been in 6 teams in the last 8 years, so he has more possibilities of having more star teammates. This is measured with “Degree” attribute. Also, it is possible to determine which players have had better squads during their career dividing the attribute “Degree with weights” by the number of years this player has been in the league. As an example, Kevin Durant has the highest value for “Degree with weights” attribute with 24, and he has played for 11 years in the league, resulting in a little bit more than two star teammates per season. However, the top player in this list is Draymond Green, who has 18 in 7 years in the league, a 2.57 coefficient. This makes sense, as Draymond Green has been in Golden State Warriors since he entered in the NBA, the best team in the last five years and, probably, one of the best squads in the entire NBA history. Finally, it can be extracted the fact that it is possible to visualize a clustering of any type of data, having as only requirement a connection between the different nodes using Gephi and its layouts.

REFERENCES

[1]: https://www.washingtonpost.com/graphics/2018/sports/nba-top-100-players-2018

[2]: https://www.basketball-reference.com/