Visualize Relations Between Papers With Gephi-Toolkit

Ruiji Wang

Shanghai Jiao Tong University

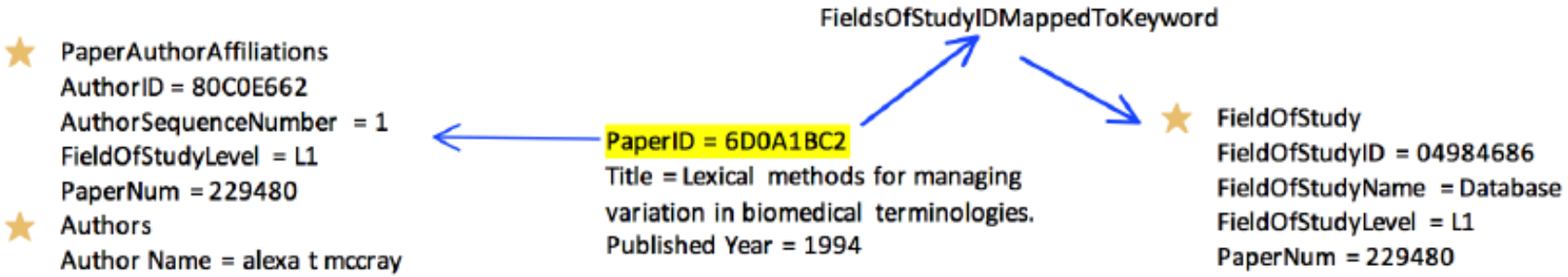
EE327

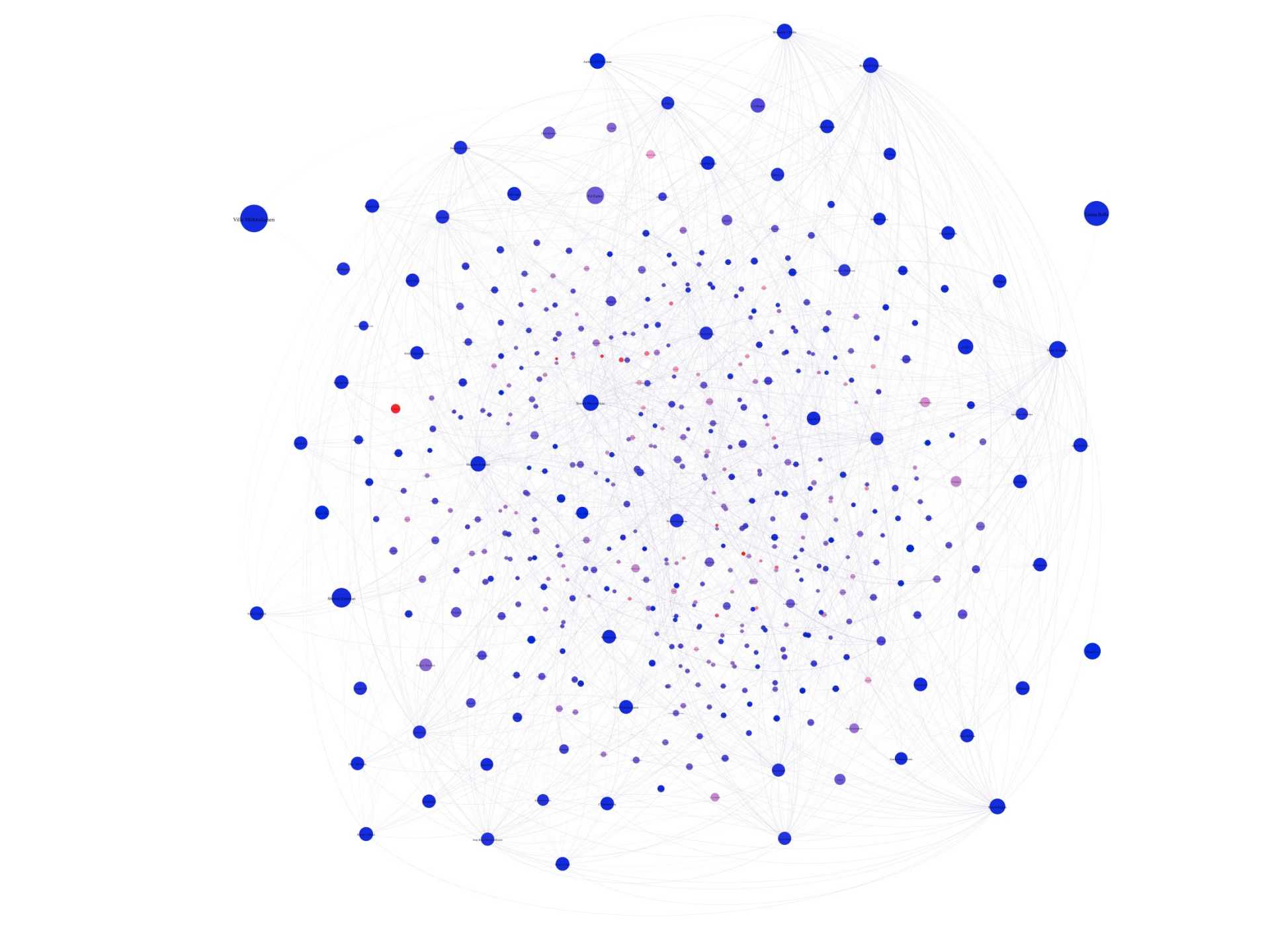
May 25, 2017

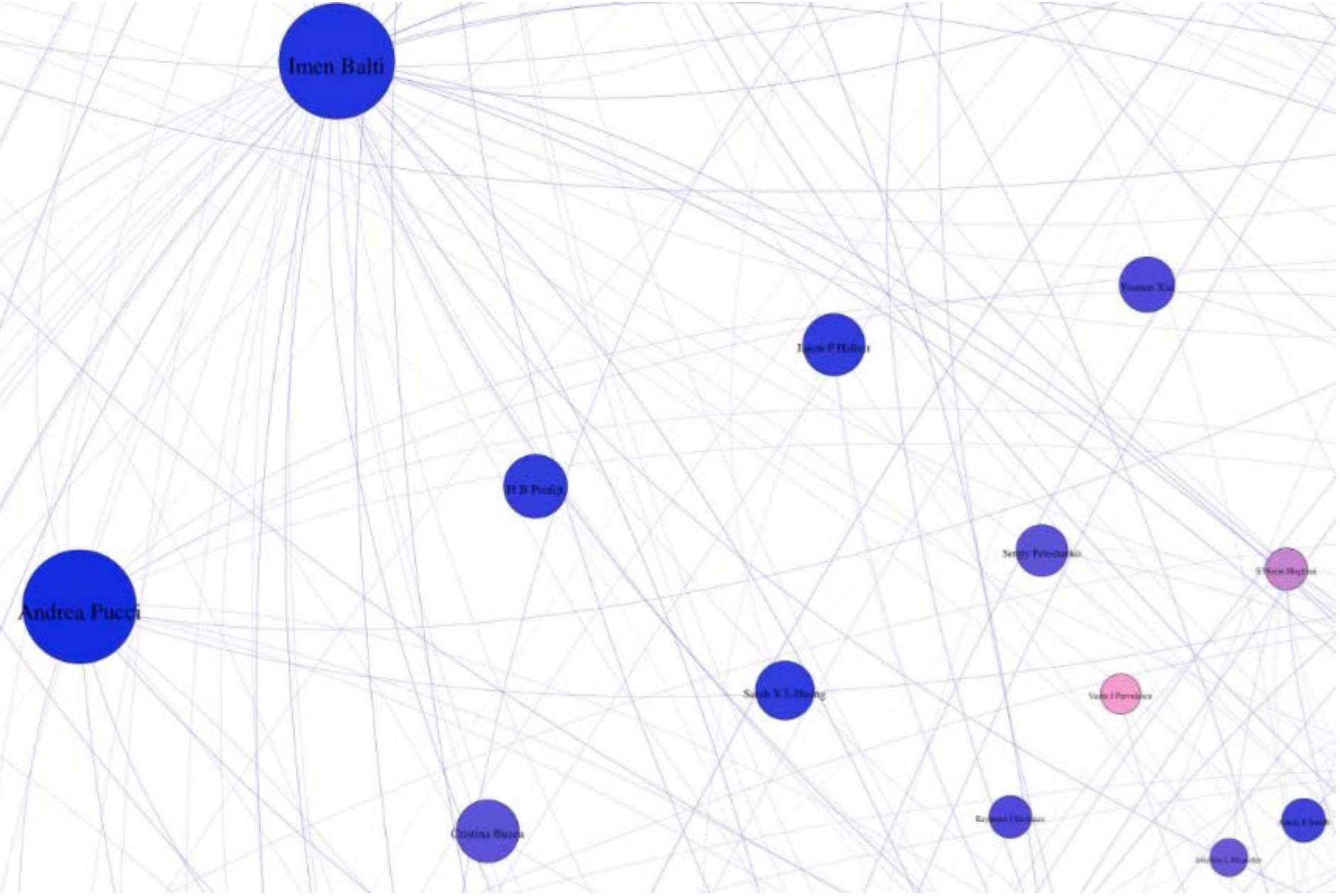
Overview

Objective

To find out the relation between papers in the same study field and use Gephi-Toolkit to visualize the result.



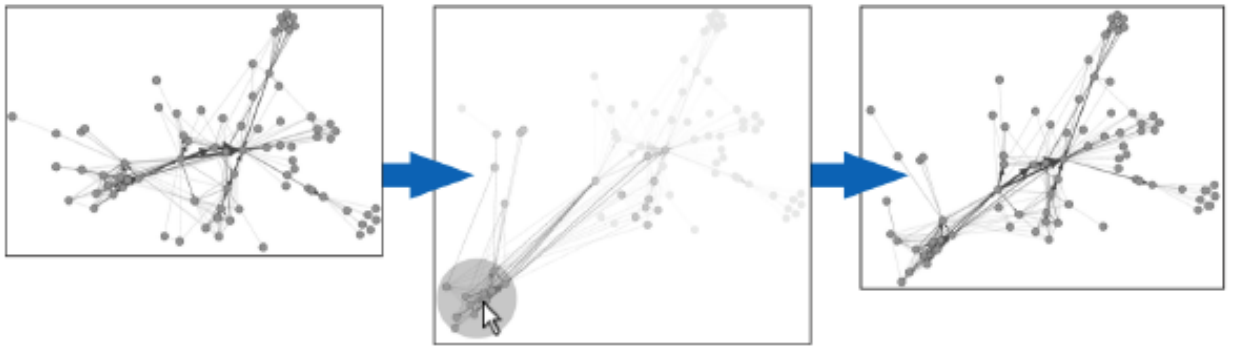

Ville Mikkonen


Imen Balti
Andrea Pucci

Introduction Of Tools

Gephi Gephi is the leading visualization and exploration software for all kinds of graphs and networks. Gephi is open-source and free.

Gephi-Toolkit The toolkit is just a single JAR that anyone could reuse in new Java applications and achieve tasks that can be done in Gephi automatically. The ability to use Gephi features like this in other Java applications boost possibilities and promise to be very useful.



Introduction Of Tools

MySQL MySQL is an open-source relational database management system (RDBMS). In this paper I will retrieve papers' information using MySQL.

mysql-connector-java The ability to use MySQL features like this in other Java applications boost possibilities and promise to be very useful.

Android Studio In this project I use Android Studio to write code in Java. Of course Eclipse and Netbeans are also good to use.

Blocks of Implementation

Database retrieval

I do database retrieval with the help of mysql-connector. I retrieve papers' information from database and find out the relations using a few MySQL sentences.

Generate GML files

According to the result of database retrieval, I can describe the relations between papers and write them into a GML file which contains the relations and some paper informations.

Generate SVG files

According to the GML files, I can draw images with the help of Gephi-Toolkit. At last SVG images shows the relations between papers in a pretty elegant way.

Generate GML files

Example (GML Nodes)

```
node [
  id "7D78A2BB"
  label "Haifeng Gao "
  reference "350"
  year 2009
  field "008D7A70|03D42FD8|08DEC03B|03905139"
]
```

Example (GML Edges)

```
edge [
  source "7D324A51"
  target "78DB16AF"
]
```

Generate SVG files

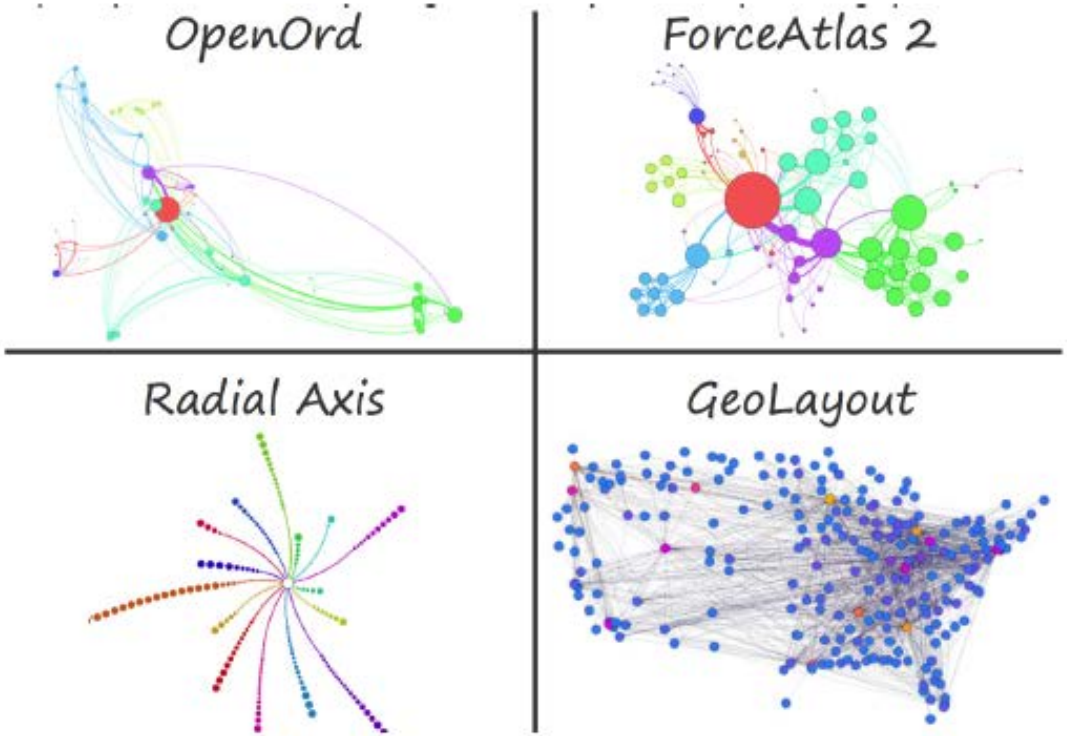
GML $\rightarrow$ SVG:

According to the GML files, I can draw images with the help of Gephi-Toolkit. At last SVG images shows the relations between papers in a pretty elegant way.

STEPS:

1. Init a project and therefore a workspace
2. Get models and controllers for this new workspace which will be useful later
3. Import GML file and data to GraphAPI
4. Filter and create new workspace
5. Calculate Nodes position and colors and Edge weight
6. Set layout and export

Gephi Layouts(I)



Gephi Layouts(II)

Layouts	Emphasis	Complexity
OpenOrd	Divisions	O(N*log(N))
ForceAtlas2	Complementarities	O(N*log(N))
Radial Axis	Ranking	O(N)
GeoLayout	Geographic Repartition	O(N)
Circular	Ranking	O(N)
Noverlap	Graphic Adjustement	—

Table: Different Layouts

Layout specification

Example (Part of code)

```
//ForceAtlas2
fa2Layout.setEdgeWeightInfluence(1.0);
fa2Layout.setGravity(1.0);
//Increase "scaling" to make the graph sparser
fa2Layout.setScalingRatio(2.0);
fa2Layout.setBarnesHutTheta(1.2);
fa2Layout.setJitterTolerance(0.1);

//NoverlapLayout
//Increase the ratio and margin for more spacing nodes
layout2.setMargin(20.0);
layout2.setRatio(2.0);
//Reduce the speed to increase quality
layout2.setSpeed(4.0);
```

Result Analyse(I)

Node Size

Node Size represents the importance of Paper which considers the paper's reference count and published year.

Theorem

$$NodeSize = \sqrt{0.5R + 0.5R \times e^{\frac{Y-2015}{15}}}$$

Notations:

$$R = Reference\ Count$$

$$Y = Published\ Year$$

Result Analyse(II)

Node Color

Node Color is only related to papers' published year.

Theorem

$$Newest\ Paper = Red\ (Color(0xFF0000))$$

$$Oldest\ Paper = Blue\ (Color(0x0426E2))$$

Result Analyse(III)

Edges

There is an edge between two nodes means that those two have one same citation field. So more edges means closer connections.

The End