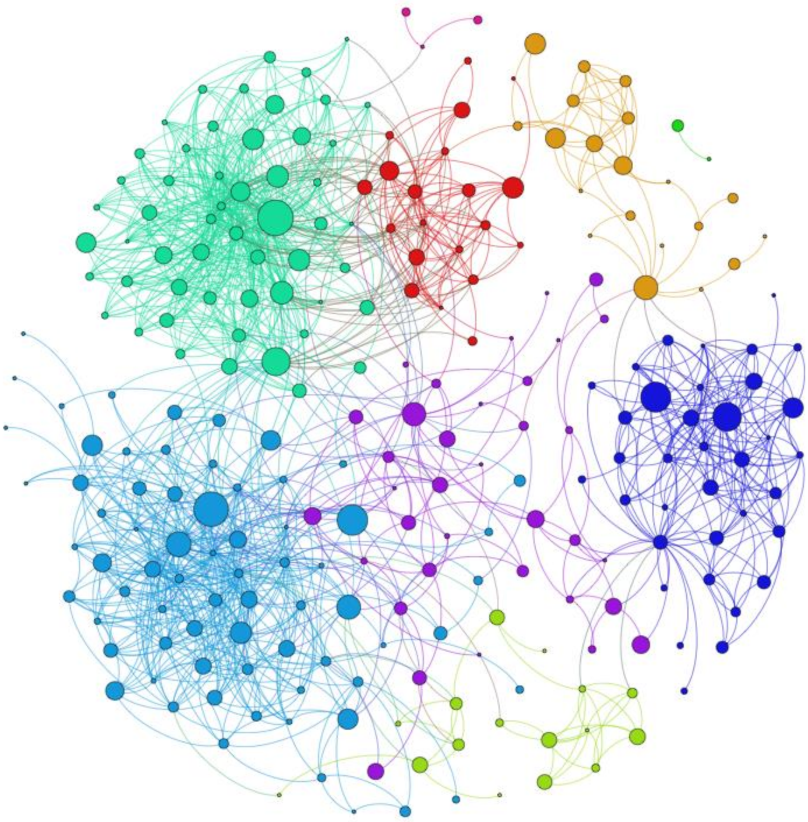
Homework II in
Social Network Analysis

From raw data to temporal graph structure exploration



Anastasios Theodorou

Master of Science in Business Analytics Student

(AM: p2822007)

Date: June 20th, 2021

Contents

1. DBLP co-authorship graph

After downloaded the compressed file “authors.csv.gz” we used Unix Tools in order to filter out data, which were above 5 years old and did not relate to those conferences: *"CIKM, KDD, ICWSM, WWW, IEEE-BigData”*. So, we selected the remaining 8727 records (command used: “*wc -l auth.csv*”). Command used for this purpose:

```
zcat authors.csv.gz | grep ",CIKM,\|,KDD,\|,ICWSM,\|,WWW,\|,IEEE BigData" | awk
'{if($1 >= 2016) print $0}'   > auth.csv
```

It is worth mentioning that in the year 2021 there were not any record referring to those conferences, because they probably have not been organized yet. So, we took the years from 2016 to 2020. We used the below commands in order to create a file for each of the wanted years.

- ```
  cat auth.csv | awk '{if ($1 >= 2016 && $1 < 2017) print $0 }' | sed -e 's/ /_/g' |
  sed ':a;s/^\(\([^"]*,\?\|"[^",]*",\?\)*"[^",]*\),/\1 /;ta;s/  */ /g' | cut -d "," -f4-  | sed
  -e 's/"//g; s/ /,/g' > auth2016.csv
  ```

- ```
  cat auth.csv | awk '{if ($1 >= 2017 && $1 < 2018) print $0 }' | sed -e 's/ /_/g' |
  sed ':a;s/^\(\([^"]*,\?\|"[^",]*",\?\)*"[^",]*\),/\1 /;ta;s/  */ /g' | cut -d "," -f4-  | sed
  -e 's/"//g; s/ /,/g' > auth2017.csv
  ```

- ```
  cat auth.csv | awk '{if ($1 >= 2018 && $1 < 2019) print $0 }' | sed -e 's/ /_/g' |
  sed ':a;s/^\(\([^"]*,\?\|"[^",]*",\?\)*"[^",]*\),/\1 /;ta;s/  */ /g' | cut -d "," -f4-  | sed
  -e 's/"//g; s/ /,/g' > auth2018.csv
  ```

- ```
  cat auth.csv | awk '{if ($1 >= 2019 && $1 < 2020) print $0 }' | sed -e 's/ /_/g' |
  sed ':a;s/^\(\([^"]*,\?\|"[^",]*",\?\)*"[^",]*\),/\1 /;ta;s/  */ /g' | cut -d "," -f4-  | sed
  -e 's/"//g; s/ /,/g' > auth2019.csv
  ```

- ```
  cat auth.csv | awk '{if ($1 >= 2020 && $1 < 2021) print $0 }' | sed -e 's/ /_/g' |
  sed ':a;s/^\(\([^"]*,\?\|"[^",]*",\?\)*"[^",]*\),/\1 /;ta;s/  */ /g' | cut -d "," -f4-  | sed
  -e 's/"//g; s/ /,/g' > auth2020.csv
  ```

Explanation:

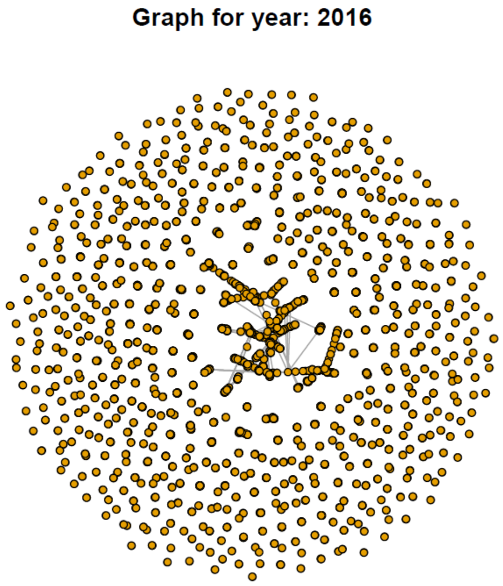
1. **cat auth.csv:** with this command we read the whole file created earlier
2. **awk '{if ($1 >= 2016 && $1 < 2017) print $0 }':** we select only the year we want
  ```

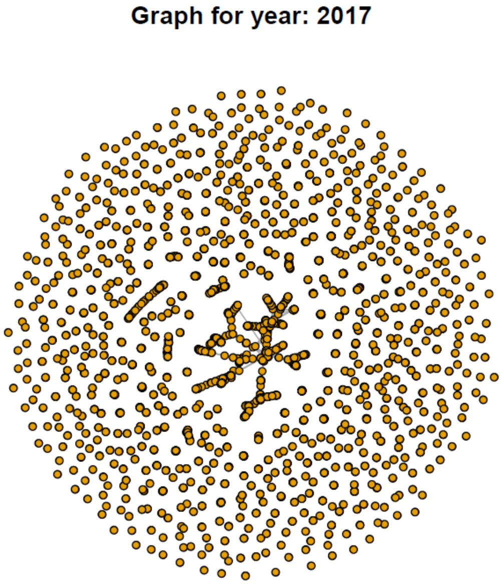
3. **sed -e 's/ /_/g':** we replace the spaces with underscores. This helps us in order not to lose any value of authors that contained both name and surname.
4. **sed ':a;s/^\(\([^"]*,\?\|"[^",]*",\?\)*"[^",]*\),/\1 /;ta;s/ */ /g':** with this command we select the values inside quotes and replace the commas with spaces, we repeat this for every line no matter how many quotes it has. In the end we remove the double spaces that has been created.
5. **cut -d "," -f4- :** we keep only the *authors* field.
6. **sed -e 's/"//g; s/ /,/g':** we delete the quotes and replace the spaces with commas so as that can be read later in R.
7. **> auth2016.csv:** we save into a *".csv"* the remaining outcome.

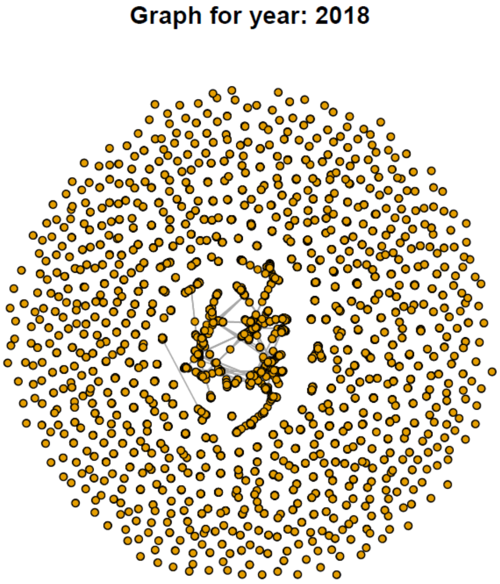
We followed the above pipeline, because there were records that contained as title of papers both commas or/and even quotes and that is why the system was confused in separation of every field. With the above procedure we took only the authors of every paper for every year, in order to use these files in R and extract our graphs. In order to check if these files contained all the data created first, we counted each line and found that in the end none of the records were missing (8727 papers).

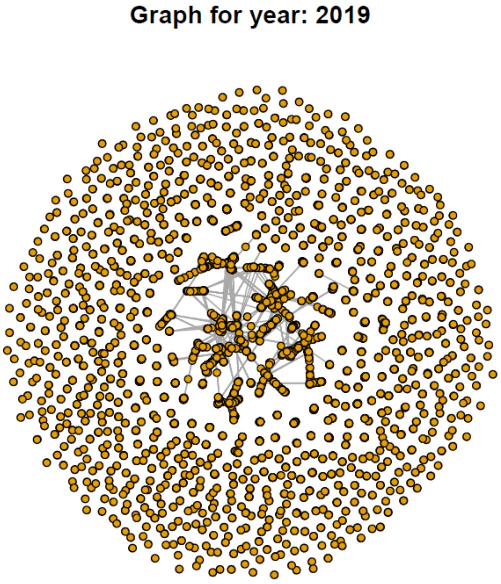
Later, we used R (R-file: *"create_files.R"*) to create files with the format we want (*author_from*, *author_t*o and their *weight*). So, we imported the files created earlier in Unix Tools and we designed a data-frame for each one of them. Before doing this, we observed through MS Excel that some records contained more than 20 authors that cooperated in one paper for a conference. That is why, in each initialization we had to use 25 columns, because if we had not initialized the number of columns, then some records that had many authors would have placed in more than 2 rows by default, something which is wrong. After that, we did some cleaning in the data (remove the NAs and those records that only one author has written a paper) and finally we created a function, which designs the graphs in the format we want. After checking for duplicates and cases where one pair of authors have been inserted twice [e.g. (author1, author2,4) & (author2, author1, 1)], we extracted the results in 5 files concerning the year we are interested in *("data2016.csv", "data2017.csv", "data2018.csv", "data2019.csv", "data2020.csv"*).

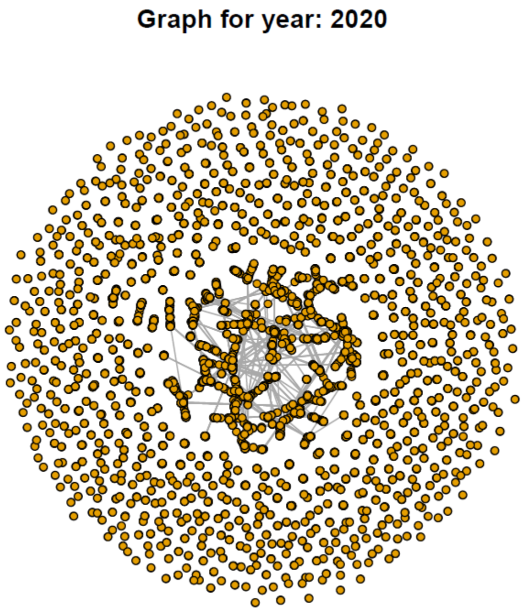
The files created earlier contained networks that can be observed in the below graphs:


Graph for year: 2016


Graph for year: 2017


Graph for year: 2018

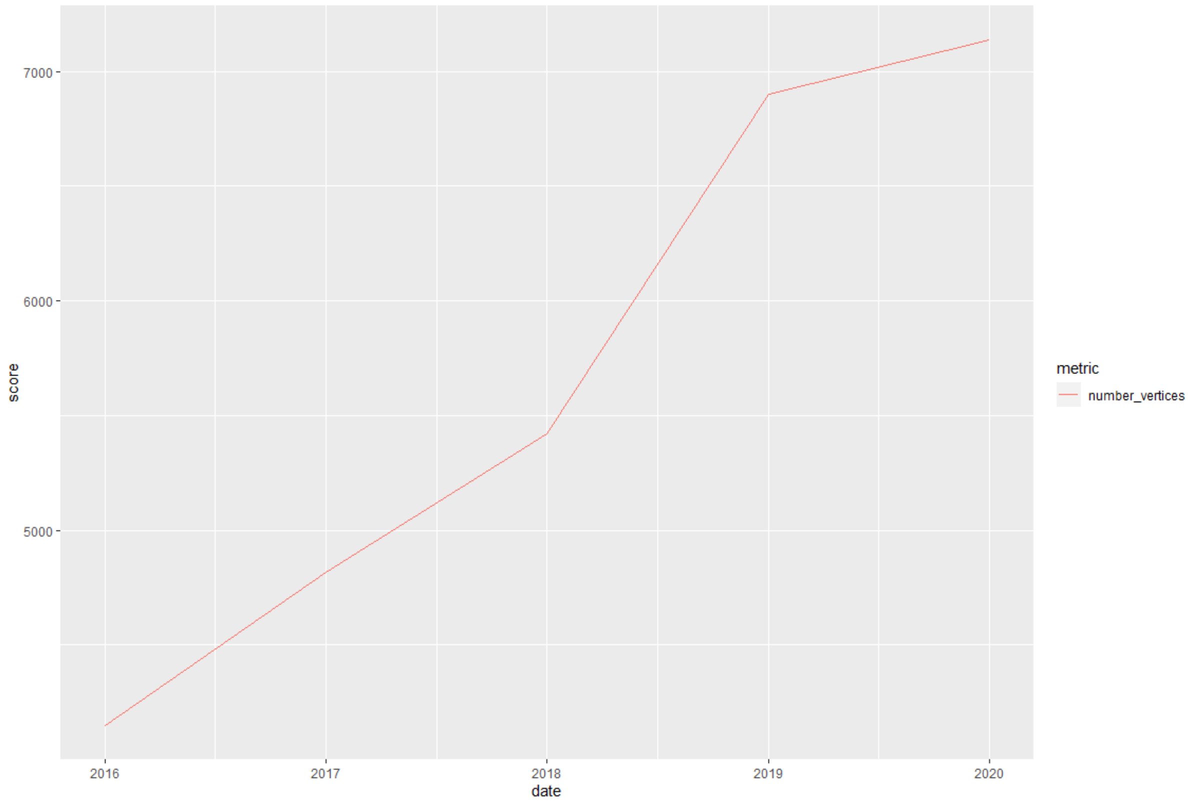

Graph for year: 2019


Graph for year: 2020

2. Average degree over time

Next, we wrote some code in R (R-file: "*hw2.R*") in order to explore the 5-year evolution of each one of the below metrics. So, we designed plots to visualize them:

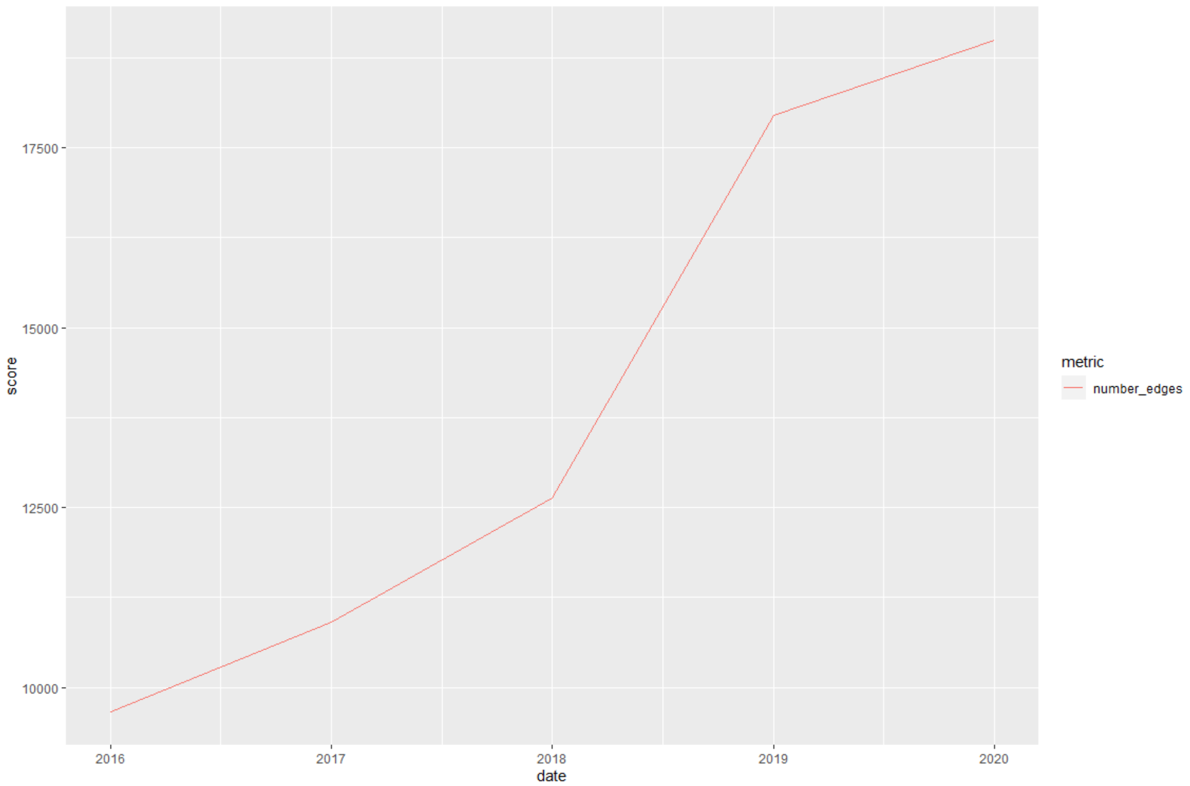
- **Number of vertices**



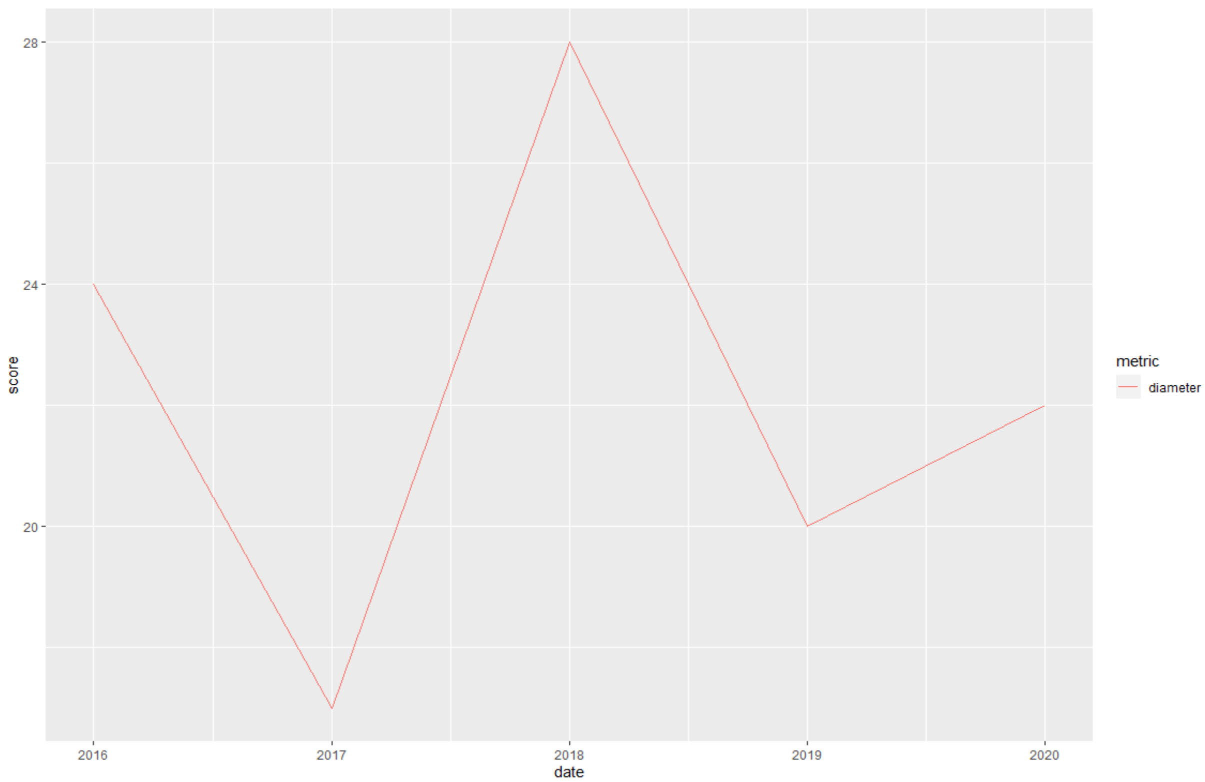
In this plot we can observe that as the years pass so the number of vertices increase. Because of the fact that each vertex represents an author, we can conclude that in the period of the last 5 years more and more authors participate in the writing of the papers.

- **Number of edges**

As we can see in the below plot the number of edges has nearly the same increase as the plot with the number of nodes. So, we can infer that more and more authors cooperate with each other in order to write papers. This was expected because as the authors multiply, so the papers and as a result the cooperation between them escalate.

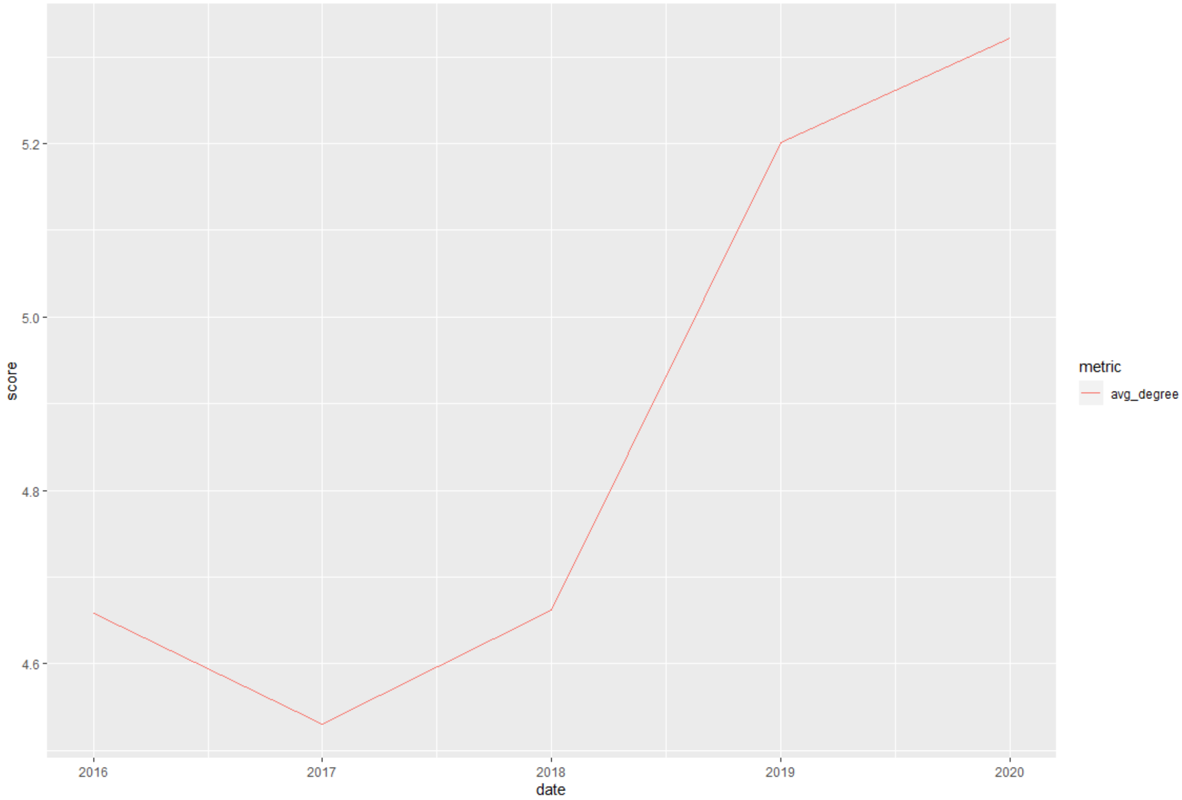


- **Diameter of the graph**



In the above plot we can observe how the diameter of each graph evolve over the years. The diameter of each graph has not the same change as the other metrics had. We can observe that the peak of it was in 2018 having papers written by 28 authors (!) and the minimum point was in 2017. The diameter in our case represents the maximum cooperation between the authors for the aim of writing a paper. That is why, we can conclude that 2017 was a year when, although we had a satisfying number of papers (from the first 2 plots) none of them was written by a large number of authors.

- **Average degree**



The final plot shows us the variation of the average degree in the last 5 years. From this plot we can conclude that here there is big fluctuation with the peak of average degree of the nodes being found in 2020 and the minimum point in 2017. If we combine the results have been found earlier, in this latter year the majority of the authors wrote the papers in small teams and not many authors cooperated with each other to complete the papers presented in the conferences we investigate. Also, it is very interesting the fact that in 2018, although we found the biggest diameter in the graph, the average degree of the authors was pretty small. Finally, in the years 2019 and 2020 we can assume that all the authors have been cooperated with more than 5 other authors on average.

3. Important nodes

Now, we will find the 10 most important nodes of each graph and we will see the 5-year evolution regarding degree and pageRank.

As far as the degree is concerned first, we saw the top-10 authors with the highest degree, as the below R output indicates.

```
$`2016`
      Philip_S._Yu     Jiawei_Han_0001       Hui_Xiong_0001         Jieping_Ye Naren_Ramakrishnan       Yi_Chang_0001
                46                  41                   39                 32                 32                  31
        Jiebo_Luo          Rayid_Ghani        Chang-Tien_Lu     Yannis_Kotidis
                29                  28                   25                 25

$`2017`
      Philip_S._Yu     Jiawei_Han_0001       Hui_Xiong_0001      Yi_Chang_0001 Claudio_Rossi_0003      Heng-Tze_Cheng
                44                  42                   38                 32                 32                  31
     Zakaria_Haque        Mustafa_Ispir      Clemens_Mewald       Martin_Wicke
                31                  31                   31                 31

$`2018`
   Philip_S._Yu Jiawei_Han_0001          Kun_Gai  Wenwu_Zhu_0001 Chao_Zhang_0014   Jure_Leskovec   Jing_Gao_0004
             70              37               35              28              27              27              27
  Xing_Xie_0001    Haifeng_Chen      Qi_Liu_0003
             26              25               25

$`2019`
     Philip_S._Yu Weinan_Zhang_0001    Hui_Xiong_0001        Jieping_Ye     Jie_Tang_0001   Jiawei_Han_0001
               69                59                49                43                39                37
     Yong_Li_0008      Enhong_Chen      Jingren_Zhou          Jian_Pei
               36                36                35                35

$`2020`
   Jiawei_Han_0001       Hongxia_Yang       Hui_Xiong_0001         Xiuqiang_He            Ji_Zhang       Peng_Cui_0001
                68                 43                   42                  41                  40                  39
Christos_Faloutsos       Wei_Wang_0010          Jieping_Ye        Ruiming_Tang
                38                 38                   37                  35
```

Then, we found from all the years the unique top authors based on their degrees and investigated their evolution over the years. These results can be shown below:

	y2016	y2017	y2018	y2019	y2020
Philip_S._Yu	46	44	70	69	27
Jiawei_Han_0001	41	42	37	37	68
Hui_Xiong_0001	39	38	0	49	42
Jieping_Ye	32	25	24	43	37
Naren_Ramakrishnan	32	15	9	10	0
Yi_Chang_0001	31	32	13	8	0
Jiebo_Luo	29	26	17	5	7
Rayid_Ghani	28	0	20	0	2
Chang-Tien_Lu	25	10	5	4	2
Yannis_Kotidis	25	0	0	0	18
Claudio_Rossi_0003	0	32	0	0	0
Heng-Tze_Cheng	0	31	0	0	23
Zakaria_Haque	0	31	0	0	0
Mustafa_Ispir	0	31	0	0	0
Clemens_Mewald	0	31	0	0	0
Martin_Wicke	0	31	0	0	0
Kun_Gai	0	6	35	23	27
Wenwu_Zhu_0001	10	3	28	26	5
Chao_Zhang_0014	5	23	27	9	34
Jure_Leskovec	12	26	27	21	17
Jing_Gao_0004	20	14	27	11	3
Xing_Xie_0001	13	10	26	18	7
Haifeng_Chen	10	13	25	0	13
Qi_Liu_0003	11	15	25	27	13
Weinan_Zhang_0001	12	13	17	59	34
Jie_Tang_0001	19	9	10	39	13
Yong_Li_0008	0	13	8	36	32
Enhong_Chen	16	15	25	36	26
Jingren_Zhou	0	0	7	35	16
Jian_Pei	20	0	17	35	16
Hongxia_Yang	0	4	22	28	43
Xiuqiang_He	0	0	0	0	41
Ji_Zhang	0	0	0	4	40
Peng_Cui_0001	11	10	18	34	39
Christos_Faloutsos	23	14	16	11	38
Wei_Wang_0010	5	0	9	18	38
Ruiming_Tang	0	0	0	5	35

Finally, for this metric we found the top 10 authors with the biggest degree on average, in order to observe who cooperated with many other authors for writing papers.

```
 Philip_S._Yu     Jiawei_Han_0001      Hui_Xiong_0001         Jieping_Ye  Weinan_Zhang_0001         Enhong_Chen
         51.2                45.0                33.6               32.2               27.0                23.6
Peng_Cui_0001       Jure_Leskovec Christos_Faloutsos    Chao_Zhang_0014
         22.4                20.6                20.4               19.6
```

After completed the investigation of the degree metric we have a lot to observe. To begin with, from a first look we can see a huge difference between the top author (degree of *Philip S. Yu*: 70) and all the others in the year 2018 (lower than 37). Furthermore, we can observe that if an author has cooperated with others for writing papers in the first year, there is high possibility doing the same in the next years. Some authors who have been seen in one year it is not necessary to appear in the next years as well. Some of them have been collaborated with many other authors in one year and have been in the top-10, but other years have written nothing (e.g. *Zakaria Haque*, *Mustafa Ispir* etc). Also, the top-10 authors for all the years have written great deal of papers across the years. It is worth saying that *Philip S. Yu* collaborates with more than 51 authors every year.

The same procedure was followed for finding the top-10 nodes based on the PageRank value. So, in this case we have the below outputs in R (***1st picture:*** top-10 nodes for each year, ***2nd picture:*** the 5-year evolution of the top nodes and ***3rd picture:*** the top-10 nodes across the year based on average pageRank).

```
$`2016`
      Philip_S._Yu     Hui_Xiong_0001    Jiawei_Han_0001         Jiebo_Luo         Jieping_Ye      Yi_Chang_0001
      0.0017288334       0.0014581015       0.0014119510      0.0013099364       0.0010027077       0.0009601005
     Hanghang_Tong Christos_Faloutsos    Maarten_de_Rijke       Jiliang_Tang
      0.0009272920       0.0009216757       0.0009158533      0.0009155034

$`2017`
   Philip_S._Yu Jiawei_Han_0001  Hui_Xiong_0001    Jure_Leskovec       Jiebo_Luo   Hanghang_Tong    Jiliang_Tang
   0.0014558956    0.0013585699    0.0010997688     0.0010681579    0.0009454158    0.0009285808    0.0007750644
  Yi_Chang_0001 Chao_Zhang_0014    Ingmar_Weber
   0.0007711858    0.0007510406    0.0007208090

$`2018`
   Philip_S._Yu Jiawei_Han_0001    Jure_Leskovec   Wenwu_Zhu_0001 Chao_Zhang_0014   Xing_Xie_0001   Jing_Gao_0004
   0.0019791353    0.0009293404    0.0008745413     0.0007835747    0.0006769059    0.0006257594    0.0006254102
   Martin_Ester  Yiqun_Liu_0001         Kun_Gai
   0.0006195914    0.0006138022    0.0006124228

$`2019`
     Philip_S._Yu     Hui_Xiong_0001 Weinan_Zhang_0001         Jieping_Ye      Hanghang_Tong    Jiawei_Han_0001
     0.0015868736       0.0009631867      0.0008766185       0.0007254176       0.0007020227       0.0006854590
    Peng_Cui_0001      Jie_Tang_0001       Enhong_Chen     Gerhard_Weikum
     0.0006573255       0.0006516757      0.0006376697       0.0006256466

$`2020`
      Jiawei_Han_0001          Hui_Xiong_0001          Hongxia_Yang Elke_A._Rundensteiner           Yong_Li_0008
         0.0010635357            0.0007588095          0.0007271062          0.0006971025           0.0006808755
           Jieping_Ye           Peng_Cui_0001           Xiuqiang_He            Ji-Rong_Wen           Jiliang_Tang
         0.0006787663            0.0006521734          0.0006454135          0.0006438328           0.0006410937
```

	y2016	y2017	y2018	y2019	y2020
Philip_S._Yu	0.00173	0.00146	0.00198	0.00159	0.00050
Hui_Xiong_0001	0.00146	0.00110	0.00000	0.00096	0.00076
Jiawei_Han_0001	0.00141	0.00136	0.00093	0.00069	0.00106
Jiebo_Luo	0.00131	0.00095	0.00059	0.00015	0.00018
Jieping_Ye	0.00100	0.00060	0.00060	0.00073	0.00068
Yi_Chang_0001	0.00096	0.00077	0.00033	0.00019	0.00000
Hanghang_Tong	0.00093	0.00093	0.00056	0.00070	0.00048
Christos_Faloutsos	0.00092	0.00057	0.00056	0.00032	0.00051
Maarten_de_Rijke	0.00092	0.00032	0.00053	0.00056	0.00032
Jiliang_Tang	0.00092	0.00078	0.00045	0.00032	0.00064
Jure_Leskovec	0.00071	0.00107	0.00087	0.00045	0.00033
Chao_Zhang_0014	0.00019	0.00075	0.00068	0.00016	0.00053
Ingmar_Weber	0.00056	0.00072	0.00032	0.00015	0.00020
Wenwu_Zhu_0001	0.00041	0.00013	0.00078	0.00054	0.00008
Xing_Xie_0001	0.00057	0.00029	0.00063	0.00041	0.00014
Jing_Gao_0004	0.00075	0.00045	0.00063	0.00024	0.00008
Martin_Ester	0.00042	0.00037	0.00062	0.00014	0.00035
Yiqun_Liu_0001	0.00037	0.00037	0.00061	0.00057	0.00028
Kun_Gai	0.00000	0.00021	0.00061	0.00037	0.00033
Weinan_Zhang_0001	0.00054	0.00046	0.00035	0.00088	0.00045
Peng_Cui_0001	0.00048	0.00037	0.00060	0.00066	0.00065
Jie_Tang_0001	0.00060	0.00030	0.00032	0.00065	0.00024
Enhong_Chen	0.00060	0.00041	0.00056	0.00064	0.00048
Gerhard_Weikum	0.00074	0.00063	0.00034	0.00063	0.00000
Hongxia_Yang	0.00000	0.00023	0.00059	0.00049	0.00073
Elke_A._Rundensteiner	0.00024	0.00045	0.00041	0.00041	0.00070
Yong_Li_0008	0.00000	0.00042	0.00023	0.00062	0.00068
Xiuqiang_He	0.00000	0.00000	0.00000	0.00000	0.00065
Ji-Rong_Wen	0.00000	0.00021	0.00015	0.00000	0.00064

Philip_S._Yu	Jiawei_Han_0001	Hui_Xiong_0001	Jieping_Ye	Hanghang_Tong	Jure_Leskovec
0.001452	0.001090	0.000856	0.000722	0.000720	0.000686
Jiebo_Luo	Jiliang_Tang	Christos_Faloutsos	Peng_Cui_0001		
0.000636	0.000622	0.000576	0.000552		

From the above R outputs, we can draw many useful conclusions. First of all, if we compare the results between degree and pageRank, we can locate one big difference in the length of the top unique nodes across the years. In the first case the length was 37 and in the second was 29. This means that as far as the pageRank is concerned there were many nodes that each year are repeated and being in the top-10. Also, another interesting finding is that if we see the intersection between the final results from the first and second case, we can find only 7 out of 10 authors (*Philip_S._Yu, Jiawei_Han_0001, Hui_Xiong_0001, Jieping_Ye, Jure_Leskovec, Christos_Faloutsos, Peng_Cui_0001*). This means that although some authors have big average degree, they are not so important.

As far as the pageRank as a metric alone is concerned, we can see the 5-year evolution of the top-10 most important nodes between the years, and we can conclude that there is not big variation of them across the years. As we observed in the first case of degree and here the top-10 authors have been repeated almost every year, like *Philip S. Yu and Jiawei Han 001*.

4. Communities

In the last task for community detection, we first implemented the fast greedy clustering, infomap clustering and the Louvain clustering in each of the 5 graphs. After executed the appropriate code in R we observed that the fast greedy clustering cannot be computed for the year 2019. This happened because in this year there are multiple edges for some nodes and that is why this algorithm cannot be calculated.

Moreover, from the executing of the above algorithms we can draw conclusions for their precision with the help of the modularity metric. This metric is a measure of the strength of the communities in one graph. If a network has high modularity, then it has dense connections between nodes within the same community and sparse connections between nodes of other communities. So, we can assume that this metric represents how good the communities have been separated. It is worth mentioning that this metric is not so powerful in finding small communities. So, in our case we have:

Year / Clustering Algorithm	Fast Greedy	Infomap	Louvain	Comparison (Infomap – Louvain)
2016	0.98	0.96	0.98	0.366
2017	0.98	097	0.99	0.346
2018	0.98	0.96	0.98	0.381
2019	-	0.94	0.98	0.621
2020	0.96	0.93	0.97	0.686

In the above table we also added the comparison between the infomap and Louvain algorithms for every year. We did so, because we wanted to see the distance between the communities, after implementing different algorithms. In this comparison we observe many variations in the value across the years. Low values mean short distance between communities and high values mean the opposite.

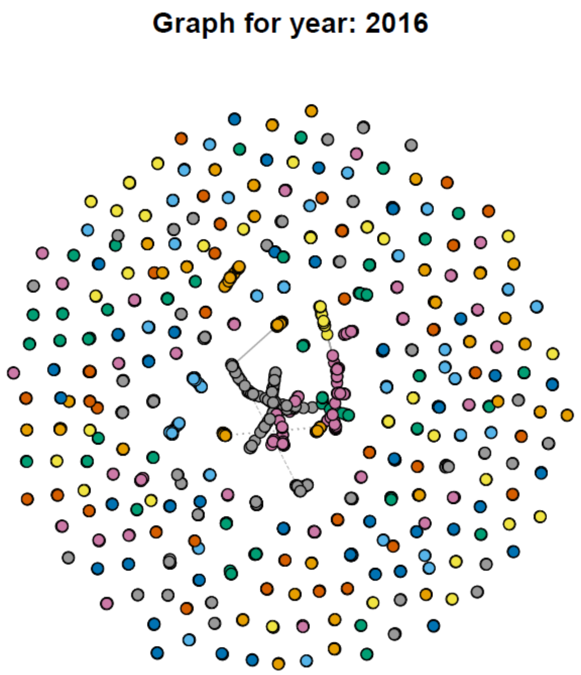
Now, for the next task we picked the Louvain clustering algorithm and a random user (in our case the *"Meng_Jiang_0001"*) that appears in all 5 graphs, and we found the evolution of the communities he belongs to. So, we found the below tables:

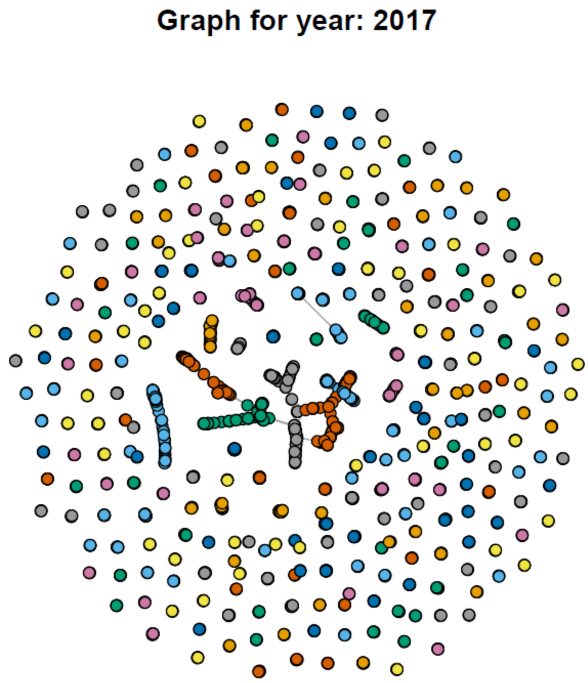
Date	Belongs to community	Length of the community
2016	559	83
2017	683	121
2018	664	86
2019	354	114
2020	351	60

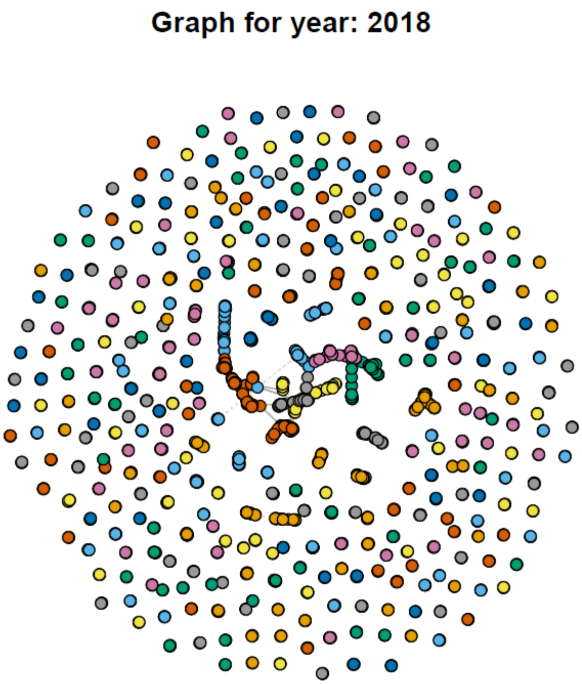
Intersection of the years	Common authors	Total
2016 -2017	Quan_Yuan_0001, Jingbo_Shang, Adit_Krishnan, Aravind_Sankar, Shi_Zhi, Honglei_Zhuang, Jisu_Kim	7
2016-2018	Jinglan_Liu, Jinjun_Xiong, Meng_Jiang_0001, Maryam_Karimzadehgan, Zhen_Qin_0002	5
2016-2019	Chao_Huang	1
2016-2020	-	0
2017-2018	Vishrawas_Gopalakrishnan	1
2017-2019	-	0
2017-2020	Miao_Lu	1
2018-2019	-	0
2018-2020	-	0
2019-2020	-	0

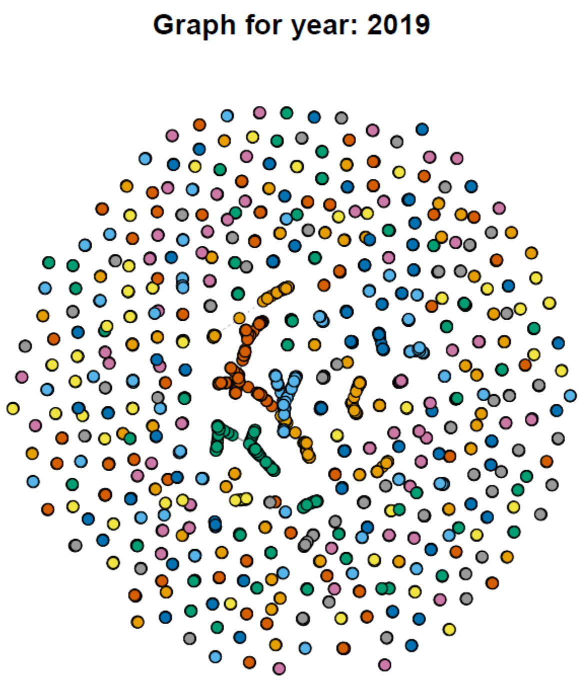
In these tables it is obvious that there are some similarities between the communities this user belonged to across the years. We can observe that his communities all over the years, as far as their length is concerned, did not differentiate a lot. Each one of them contained between 60 to 121 authors with the majority of the years being above 80. Moreover, in the second table we can see the evolution of this user's communities and their similarities among them. The first years (2016-2018) we can see that his communities contained many common nodes. In particular, between the years 2016 and 2017, the user had 7 common neighboring nodes and between the years 2016-2018 he had 5. In the rest of the years, he had 1 or even 0 common neighbors.

Finally, we designed 5 plots, one for each of the above graphs in order to detect their communities. We used different color to separate them. We also filtered out those vertices which created too large communities (above 100 nodes) or even too small (below 5) for having more pleasing visualization.









Graph for year: 2020