Identifying Influential Users on Instagram

Student Name: Chen, Qian

Supervisor Name: Barnaby Martin

Submitted as part of the MSc <Chen,Qian> to the

Board of Examiners, School of Computing Sciences and Engineering, Durham University

[[1]](#footnote-1)

*Abstract*

**Context**: Social media applications have become major platforms for users to share their personal lives and opinions or promote some marketing work. As more and more people and companies are using social media applications for personal and professional reasons, companies, data scientists and researchers are all trying to do some researches and infer some meaningful information from mass data on such social networks. Centrality is one of the most studied concepts in social network analysis which can measure user influence on social networks.

**Aim**: There are many measurements for calculating network centrality depends on different features. The purpose of the project is to identify influential users by different measures, classify and compare them so that those measures can be reasonably applied in practice according to relevance criteria.

**Method**: The first experiment is based on the “following” relationship on Instagram network. In this experiment, except for traditional centrality metrics, like Degree Centrality, Closeness, Centrality, Betweenness Centrality and PageRank, some other new centralities were also implemented and were used for comparison. The second experiment is trying to figure out whether user influence can be different across different topics. TwitterRank and MentionRank were used to be implemented and compared. The third experiment is to integrate the whole process of Social Network Analysis and make a visualization. The application is a web-based tool combining functions of scraping, analyzing and visualizing.

**Result**: Those measures in the first experiment are closely related even though they are based on different ideas. However, in the second experiment, the ranking of influential users by TwitterRank and that by MentionRank shows a large difference upon fixed topics. But both of them can prove the hypothesis that user influence varies across topics. In the final experiment, user can scrape a certain number of user information by providing a valid username of Instagram and get the relationship graph and user influence ranking.

**Conclusion**: User influence varies across different ideas of measures and different topics, so it is important to make a suitable definition for user influence before selecting an influential measure.

***Keywords*** –Centrality, PageRank, Topic-sensitive Algorithems, Social Network Analysis

# **Introduction**

Social Network Analysis (SNA) emerges and develops with the appearance of Social Network Services. Social networks have four major features: rapidity, extensiveness, equality, and self-organization. Because of these features, social networks have had billions of users and has a huge effect on almost everything in people’s lives since the beginning of the internet (Noam Segev, Noam Avigdor, and Eytan Avigdor, 2018). In the 2016 U.S. presidential election, Trump made good use of Twitter as a propaganda tool and finally he won the election. Nike organized an interesting activity named NikeiD that users can design their own shoes in Instagram. This activity not only helped Nike attract more customers successfully, but also made users have a chance to be a shoe designer. In this activity, those users encouraged and delighted each other, and shared their designs and ideas in the social networks.

However, except for the positive impact to the society and economy, social networks also have some negative impacts. There is so much false and harmful information on the Internet, such as terrorist information and violent messages in Facebook and YouTube, and a lot of gossip, rumors and fake news in Twitter and Instagram, that will do harm to our society and tend to have uncontrollable consequences due to its rapid spreading feature.

In order to make good use of social networks to produce value and reduce hazards for our society, Social Network Analysis, a new kind of science, emerged. This is a subject that combines information science, mathematics, sociology, management, psychology and so on (Fang et al., 2014).

In the field of SNA, influence analysis is one of the mostly researched areas. The notion of influence plays a vital role in how companies promote and how information disseminates (Meeyoung et al., 2010). There are many successful application cases based on researching user influence on social networks, such as viral marketing (Domingos and Rechardson, 2001) (Kempe, Kleinberg and Tardos, 2003), information propagation (Golbeck and Hendler, 2006) (Gruhl, Liben-Nowell, Guha, and Tomkins, 2004) expertise recommendation (Song, Tseng, Lin, and Sun, 2006).

Instagram is a visual content sharing online social network services and it gradually becomes an important platform for influence promotion(Noam Segev, Noam Avigdor, and Eytan Avigdor, 2018). Here are the benefits of identifying influencers for different groups:

* To social network services (Instagram): Identifying influential users in Instagram can help this website get more traffic because users will create more valuable content if they want to be ranked in a high level which may attract more fans for them. At the same time, those fans will also become active users in Instagram.
* To users: Celebrities can gain the direct or indirect profit such as fame, prestige, advertising revenue, high social status and high reputation by increasing their influence. Ordinary users can also find the influencers who are interested in progressing quickly through the User Influence Rank.
* To merchants: Merchants or companies which want to promote their products or services can find influential and high-value commercial users by rating and determining users influence value and cooperate with them for brand advertisement. That would be an effective method of marketing.

However, there are many centrality measures to compute user influence. Traditional metrics, like Degree Centrality, Closeness Centrality, Betweenness Centrality and PageRank, etc. which rely on simple graph-based algorithms of a “following” relationship network graph. But there are also other many measures based on other relationship network graphs or even based on complex mathematical models. So here comes a question: whether all centrality measure can be applied well in Instagram? How can we find the relevant influencers by using suitable measures?

The key to this question is the identification of influential users in Instagram. According to different criteria or network structures, users’ value will be different, so their influence will be diverse. For example, Degree Centrality is more likely to rank user influence when the request is to spread information as soon as possible. But if the requirement is to spread information to as many people as possible, Closeness Centrality is more appropriate. Besides, upon different topics, user influence also should be varied. For instance, even though super soccer star Ronaldo has a huge number of fans in Instagram that cannot guarantee that he can achieve great promotion effects in beauty makeup field.

Therefore, the research question of the project is:

**Q1**: “How to find measures that can be worked efficiently and reasonably according to relevant criteria as close as possible to the reality in Instagram?”

**Q2**: “How to evaluate user influence across different topics in Instagram?”

The purpose of this project is to collect different influence measures, then classify, compare and contrast them. So, when there is a specific criterion and standard of the user influence, the most relevant measure can be picked to rank the user influence and identify the most relevant influencers according to some certain requirements. Meanwhile, in fact, most research about influence measures in SNA so far are based on the Twitter Network, so this research also presents what measures are suitable on the Instagram Network, what measures are a poor fit for Instagram even if they work very well on other social networks.

To successfully address the research question and achieve the goal of the project, the deliverables have been defined clearly before the implementation process:

1. **Basic**: make graphs of the “following” relationships among Instagram users and rank the user influence based on the network structure generally.

Brief:

Based on the structural features of a graph of a social network, there are several traditional methods to measure a node’s centrality in the network, including Degree Centrality, Closeness Centrality, Betweenness Centrality and PageRank. In addition, some other new centralities, K-path, K-shell and K-hop algorithms, are also implemented to calculate user influence.

* Gather thousands of users’ following information automatically and make a graph to represent the relationships among them.
* Make rankings to rank their influence based on different centrality measures and compare them.

1. **Intermediate**: analyze user influence across topics based on user behavior.

Brief:

Users’ actions can also affect user influence in a social network. In Instagram, the main behaviors are comments, replies, thumb ups, mention and tag username, so various graphs about the network relationship can be constructed based on those behavior characteristics. In this deliverable, apart from the “following” relationship graph, the “mentioning” relationship graph can also be constructed to be used to calculate user influence.

* Gather thousands of users’ posts in Instagram automatically
* Make a distribution of user ranks for three given topics based on their content of posts.

Incorporate both the users’ relationship in network structure and the users’ posts on given topics to identify topical influencers depending on different topics.

1. **Advanced**: GUI and Visualization

Produce a User Interface to provide users with a user-friendly way to operate the software. In this system, users can input a username of Instagram to scrape the profile’s following users in real time, and after that, the system can analyze the user influence and visualize it in an appropriate way.

Table 1 shows the outcome of the Project.

Table 1: Outcome OF This Project

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| measures | ? | Advantages | Disadvantages | Application |
| Degree Centrality | local | fast, easy to compute | Does not consider network topology | Find the users with high reputation |
| Betweenness Centrality | global | Considers the whole network topology | High computational complexity | Find the users acting as a bridge connecting different user groups |
| Closeness Centrality | global | Considers the whole network topology | High computational complexity | Find the users who can spread information to more users quickly |
| PageRank | global | Modelled as a user behavior | Always rank celebrities/spammers in high place | Find the users who have high-quality users |
| K-Path | k-local | Based on Betweenness Centrality and lower computational complexity | The accuracy really depends on the value of *k* | Large-scale network and consider the randomness of information dissemination |
| K-Hop | k-local | Based on Degree Centrality and considers more topology structure | The accuracy really depends on the value of *k* | Large-scale and dynamic network |
| K-Shell | global | Reveals the full hierarchy of nodes | Cannot rank user influence precisely | Classify users influence in a network |
| TwitterRank | global | Combines content analysis and the "following” relationship | More likely to rank topic-related but not highest relevant celebrities/spammers in high place | Find topic-related celebrities |
| MentionRank | global | Considers “mention” behavior of user in reality | Does not consider network following topology which is an important feature for identifying influencers | Find specialists in a filed |

The rest of the report is organized as follows: In Section 2 relevant influence analysis works in SNA will be described. In Section 3 the details of the process of the project, including the designing work, data collection, coding work and testing, will be given. And then, results of the experiment will be shown in Section 4. In Section 5, evaluation work about comparing and contrasting different algorithms, evaluating their performance and usability, and analyzing why will be explained. Finally, Section 6 will be a conclusion of the project and some suggestions for the future work.

# **Related Work**

In SNA field, influential user identification can be classified into two groups: (1) Simple Graph-Based Approaches; (2) Graph-Based Approaches with Nodal Features. In fact, some Non-Graph Based Approaches are also able to be used in measuring user influence, but those measures do not take network information into account, so they do not belong to SNA indeed.

(1) Simple Graph-Based Approaches

Refer to Graph-Based Approaches, there are three very basic and classical centrality measures:

* Degree Centrality: the number of direct neighbors of a node.
* Closeness Centrality: Sum of the minimum distance between the node and other all nodes in networks.
* Betweenness Centrality: Betweenness Centrality is used to measure the intermediating effect of a node in social networks, trying to find a bridge or a powerful hub in a network.

Degree centrality is local centrality but is less effective, because it does not consider the node network topology. Both closeness and betweenness centrality are global measures, and they could better identify important nodes but both of them are too time-consuming for dynamic large-scale network (Niu et al., 2014).

Link analysis algorithms can also be used to measure user influence- the most popular one being PageRank (Page, Brin, Motwani and Winograd, 1998). PageRank was proposed by Google to rank web pages according to their importance on the Internet at first, but later on, researchers found that such algorithm is also suitable to be used to measure users’ influence in a network because it is a very similar behavior that a user clicks links on a web page on the Internet and that a user follows a following of a profile on a network to some extent. PageRank is a random walk algorithm and in SNA, its score is the transition probability that a node follows another node.

PageRank has a problem that it is prone to put spammers at a high ranking since the repeated iterations will give much power to the highly connected nodes, no matter it is high indegree or high outdegree (Pal et al., 2016), (Gayo-Avello, 2013). Therefore, Kwak et al. (2010) found that the ranking result of PageRank is similar with the ranking of follower count in Twitter but does not has pertinence with retweet rate which is one of important indicators of influence because retweet can increase information diffusion (Meeyoung et al., 2010).

K-path Centrality (Ck) (Alahakoon et al., 2011) has similar nature of Betweenness Centrality (Cb) but it is a random algorithm approach, so it can solve the problem of Cb that the shortest path is not always the only path that influential nodes in the flow information can go through. Therefore, K-path Centrality can be seen as a more efficient way to quantify the influence of a node in the network compared with Cb. K-shell decomposition algorithm also is a popular measure to be used to identify influential users to dynamically deal with complex networks (Kitsak et al., 2010). However, Niu et al. (2014) found that K-shell is time-consuming because of its global feature and has weak recognition capability. K-hop (Niu et al., 2014) is a generalization of degree centrality, but it exploits more topological information of networks. Compared with those mentioned Graph-Based Approaches, the most prominent characteristics of K-hop is locally calculated and complexity-scalable, because other algorithms above (except K-Path) were designed for static, small-scaled networks. So, if the network is large-scale and dynamic, K-hop is the nice choice.

All of these Graph-Based Approaches are all only based on the node network structure and do not take user specific information into account, so they have obvious limitations when applying them to real-world issues because the relationships in a network is just one criteria of influence analysis (Zengin Alp and Gündüz Öğüdücü, 2018).

(2) Graph-Based Approaches with Nodal Features

An influential measure which can combine network topology and some user features would be more practical in user influence identification.

TunkRank (Tunkelang, 2009) is a variation of PageRank. TunkRank takes retweet into account when measuring user influence. It defines influence as the probability that the number of users who will view or retweet/repost the tweets/posts of the user eventually in social networks. Due to its similarity to PageRank, the accuracy of the result would be affected by link spammers, just like PageRank. Despite that, Gayo-Avello (2013) proposed that TunkRank could penalize spammers better than PageRank thought experiment.

When network analysis is combined with content(text) analysis, topic-sensitive algorithms can be generated. Topic-sensitive PageRank (TSPR) (Haveliwala, 2003) and TwitterRank (Weng et al., 2010) are both improved algorithms of PageRank with a topic-biased teleportation vector. The difference between them is that as PageRank, TSPR was also applied in ranking web pages, so it is short of pertinence about user behaviors. TwitterRank makes up for it effectively and reasonably. TwitterRank combines the nature of PageRank and content analysis in users’ posts and extract topic information from those posts by using LDA topic model. Thus, the result of TwitterRank will rank the user influence according to different topics. But the spammers problem as PageRank also exists in TwitterRank (Pal and Counts, 2011), which is the common problem of algorithms that based on PageRank.

All the mentioned algorithms above are all based on the “following” relationship graph, Xiao et al. (2013) proposed a new topic-sensitive measure to calculate user influence: MentionRank which constructs the “mentioning” relationship graph with weight that represents the times of a user mentions another user in topical posts. MentionRank does not use topic model to get topics but uses a target word to detect hashtags by using PIOLog (Probabilistic Inside-Outside Log method). So, it requires predefined a target word in advance and extracts topical posts by hashtags.

In summary, based on different definitions of user influence in social networks, there have been extensive related work to measure user influence according to relevant criteria. Most of them are based on the “following” relationship graph theory and applied in Twitter. Recently researchers also try to highlight the use of other metrics (retweet, mention, reply, favorite or like) in influencer identification, especially retweet (Riquelme and González-Cantergiani, 2016). But given the topic of this project is to identify influential users in Instagram where there is no retweet/repost metric, and because there is relatively little work on Instagram, it would be a problem that what influential measure is more suitable for Instagram.

# **Solution**

In this section, the whole process of this project will be given. Figure 1 shows the architecture of the solution.

Data Collection

Data Preprocessing

Graph-Based Measures

Basic

(network topology)

Intermediate

(Topic-sensitive)

Topic Modeling

Hashtags Community

TwitterRank

MentionRank

Advanced

(GUI)

Web-based system

Scraping

Visulization

Figure 1. Overall Process

## **Data Collection**

In order to calculate user influence in Instagram by using Simple Graph-Based Approaches, there would be an Instagram dataset which is a table of relations consisting of username and the following’s username. The data could be obtained from Instagram API before December 11, 2018, but later on the capability was disabled for improving Instagram users’ privacy and security.

For the limitation of the Instagram API, crawling pages and gathering information became the optimal choice. Python is a scripting language and easy-to-configure, in addition, it has rich fetch components access the Internet, so Python is very suitable for crawling work. Then Python became the basic programming language of this project and PyCharm, a Python IDE, became the main developing software in the project. By using Instaloader, an external library of python which is used to scraping web pages in Instagram, Instagram data could be scrapped. Data set used in this experiment comes from Instagram taken from Instagram @cristiano user, and followings of those users are collected in a breadth first manner until sufficient number of users are collected (expectancy is 2000 users at least).

However, because Instagram has set an anti-spider system to limit the machine crawling, this scraping project always failed and returned the wrong HTTP code 429 (which means Too Many Request) when crawling. Anti-spider technology is used to distinguish real user from bot on the website. Basically, anti-spider system mostly adopts the following strategies: (1) User-Agent + Refer detection (2) User account and cookie detection (3) Frequency Restriction of user behavior by same IP address. But actually, crawling behavior can be almost utterly close to real people, for example (1) Use different http headers to emulate browser environments (2) User accounts can be bought from many online shopping websites (3) Proxy IP addresses can be also scraped from some proxy websites. To put it simply, as long as a cookie pool has enough user accounts and a proxy pool has enough and validated proxy IP addresses in the scraping project, and every request is made by different account and used in different IP addresses automatically, the anti-spider system cannot detect the machine crawling. So, after the cookie pool and the proxy pool have been built, the scraping work went well and rapidly.

The data of user “following” relationship is collected and stored in SQL database. Figure 2 demonstrates the complete ERD of all collected data and Table 2 shows the Statistics about the data set.

A screenshot of a cell phone

Description automatically generated

Figure 2. ERD of the whole project

Table 2. Statistics about data sets

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| scrape | User\_info | User\_relation | Post\_info | Mention\_info |
| number | 1826 | 68918 | 146665 | 151217 |

## **Preprocessing**

A list is created to store all “following” relationships between the collected Instagram users. Each element of the list is a tuple with two user ids. First one is the user id in User\_info table, second one is one of the following user ids of that user. Create a directed graph for this data and add edges from this list. Therefore, a complete graph(network) was built with 1826 nodes and 68918 edges.

## **Centrality Measurements**

NetworkX is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks. In Netwrokx library, there are many standard graph algorithms, including Degree Centrality, Closeness, Centrality, Betweenness Centrality and PageRank. The result of those centralities is a dictionary where the key user id and the value is the score of each centrality.

Based on the created graph, three new centralities, K-Path, K-Shell and K-Hop, which have not been implemented by network, were implemented in the project.

### **K-Path Centrality Measurement**

The following Figure 3 shows the whole process of K-Path centrality measure.

Path Determination

Initialization

Accumulation

Figure 4. K-Path Centrality Measurement Process

The first step is to initialize required parameters in K-Path centrality: *α, k, T*. Where *T* is the iterative times of the paths generated, *α* ∈ [−1/2, 1/2] is a parameter that controls the tradeoff between accuracy and computation time. In this experiment, *α* is set as as **0.2**, *T* as **2*k*2*n*2-2αln*n***on this unweighted graph. *k* is a positive integer that represents the number of the most links that a message can traversal. *k* here is set as **ln(*m*+*n*)**. *n* is the number of nodes in this graph, and m is the number of edges in this graph.

After the initialization, the second process is to determinate paths for the random message. In this process, a message from a random source s ∈ V in the graph, and *ledges* ∈ [1, κ] are chosen uniformly at random. The message can go along the simple path (do not repeat) at random with *ledges* edges. After the iteration time of O(**2*k*2*n*2-2αln*n***), how many times have each node been picked up can be counted. The more times a node been gone though, the higher K-Path score the node has.

The last step is to normalize the final K-Path score of each node.

### **K-Shell Decomposition**

By the K-Shell decomposition algorithm, all nodes with degree k = 1 should be removed in the graph, and then a new graph will be appeared after the removing operation. Repeatedly, continue removing those nodes with one link in this new graph, until there is no node left with degree k = 1. Each node in this graph can be associated with one k-shell index. The k-shell index of the nodes in the first removed group is 1, and iteratively k-shell index of the next removed layer nodes is 2. As a result, the k-shell index of each node can be collected.

### **K-Hop Centrality Measurement**

In the implement of K-Hop, *k* is a parameter to control the accuracy and the computation time of the algorithm dependent different application scenarios. The greater value the *k* is, the more accuracy the result would be. *α* is the average degree of all nodes in the graph, and *ni*(*v*) is how many nodes that the path length is at most *i* from node *v*, where *i*=*1,2,3*...*k*.

So, first of all, in this experiment *k* is set to 3, and *α* is initialize as an integer number 37 (divide the number of edges *m* by the number of nodes *n*). The K-Hop Centrality score of each node can be calculated based *ni*(*v*) and *α.*

## **Topic-Sensitive Algoritms**

### **TwitterRank**

#### **Text Preprocessing and topic model**

In this stage, at first all the collected posts should be preprocessed. Posts in Instagram often contain equivocal or meaningless words or icons and can be written in many different languages. In order to make the result of TwitterRank clear and non-intrusive, the preprocessing of posts is to exclude non-English words, unprintable words, and “stop-words ” which is a manually created words list contains unmeaningful words, like “the”, “a”, “me”, “she”, “and”, “but” etc.

After the data clean and LDA topic modeling, topic distribution of documents can be calculated by LDA topic model. In this experiment, the number of topics is set to 10, and the number of iteration times is set to 100. As a result, LDA topic model will return a two-dimensional matrix named dt where dt[i][j] represents the weight of doc[i] belonging to topic[j].

#### **Calculating TwitterRank**

After gathering user “following” relationships, and the number of each user’s posts in Instagram, the probability that a user *i* is influenced by a user *j* upon a specific topic can be calculated as well. The result is also a two-dimensional matrix named Pt. Pt[i][j] will demonstrate *i* follows *j* and the value of Pt[*i*][*j*] depends on the number of topical posts *j* has posted on Instagram, and the similarity between the weight of the two users on this topic. The more topical posts, or the higher similarity, the greater Pt[i][j] value is, which means *j* has higher influence on *i* on this topic.

As a result, TRt is a list that stores every user’s influence according to each topic t. This list depends on both the value of Et and Pt where Et is a list that stores the normalized weight of attention from each user to topic t. γ is a parameter to adjust the probability of teleportation, and γ was set to 0.85 here. The algorithm is in iteration, until the Euclidean distance between the last TRt and the previous one is less than the tolerance (set as 1\*10-16 here).



### **MentionRank**

#### **Hashtag community detection.**

Unlike TwitterRank, topic should be predefined in MentionRank by a manually target word. According to this target word, topical hashtags community can be easily found in some existed websites like <https://www.shopify.com/blog/instagram-hashtags>. Three predefined topics was set in this experiment: Sport, Home, Fashion. By using the above website, the hashtags list can be easily gathered. The hashtag community of each topic lists below:

Table 5. Selected topics and relevant hashtags

|  |  |
| --- | --- |
| Topic | Hashtags |
| Fashion | #fashion, #style, #OOTD, #InstaFashion, #FashionBlogger, #Fashionista, #StreetStyle, #Stylish, #InstaStyle |
| Home | #home, #homedecor, #interiordesign, #design, #interior, #house, #architecture, #decor, #love, #photo |
| Sport | #sport, #fitness, #gym, #training, #motivation, #fit, #workout, #sports, #soccer, #football |

To simplify this algorithm, the relevance between the hashtag and the topic is initially set to 1.

#### **Calculating MentionRank**

MentionRank is also a random walk algorithm but based on “mentioning” activities. In the nature of MentionRank, a user is more likely to visit another user that he/she mentioned in his/her posts before. The author of MentionRank referred to this trainsition matrix for a certain topic as AMR. AMR (*ua,ub*) indicates the rate of user *a* mentioning user b in *a*’s topical posts where topical posts here mean that those posts contain any hashtag in hashtag community based on one topic.

After that, a list named TVc is created to store the relevance between each user and the certain topic. So, we assume that *c* is a topic here, each element of TVc here is the relevance between the user and the topic c. The relevance is calculated by the hashtags the user used in Posts.

Therefore, MentionRank can be calculated based on TVc and AMR.

## **GUI**

The advanced deliverable is to design a GUI to visualize the result. This GUI is a web application, and in order to make the operating process of the system smooth, this app integrates the different parts of the project from scraping, through analysis and visualization.

Django is the most popular web frame of python. It is based on MVC software design pattern and easy to start. More important, it is free and open source, so Django is suitable for this project.

The following is the directory structure of the project.

|-- ScrapeRealTime  
| |-- \_\_init\_\_.py  
| |-- db\_operation.py  
| |-- ins\_relationship.py  
| |-- ins\_scrape.py  
| |-- my\_centralities.py  
| |-- settings.py  
| |-- StopWords.py  
| |-- urls.py  
| |-- view.py  
| |-- wsgi.py

|-- ScrapeModel  
| |-- media  
| | |-- img  
| | |-- javascript  
| |-- migrations  
| |-- static

| | |-- css

| | |-- img  
| | |-- javascript  
| |-- \_\_init\_\_.py  
| |-- admin.py  
| |-- apps.py  
| |-- models.py  
| |-- tests.py

| |-- view.py

|-- templates  
| |-- card\_pic.html  
| |-- more\_information.html  
| |-- ranking.html  
| |--scrape\_relationship.html

`-- manage.py

Figure 5. Directory Structure of the Project

“ScrapeRealTime” directory stores all the main functions of the project, “ScrapeModel” directory is used to connect MYSQL and create tables in the database. All “html” files are in the “templates” directory.

# **Results**

This section will show the result of the project. In all result ranking tables, same username will be with the same shading so that it is clear to compare and contrast with different algorithms.

## **The result of Graph-Based approaches:**

The general ranking of Top15 influential users by centralities is shown as follows:

Table 5. The general ranking of Top15 influential users

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| rank | betweenness | closeness | degree | pagerank | k-path | k-shell | k-hop |
| 1 | cristiano | cristiano | cristiano | cristiano | cristiano | ikercasillas | cristiano |
| 2 | mattiacaldara | neymarjr | mattiacaldara | vogueparis | natgeo | cristiano | neymarjr |
| 3 | blessedmma | kingjames | 433 | natgeo | willsmith | kingjames | davidbeckham |
| 4 | paulomeixedo | davidbeckham | neymarjr | voguehommes | mattiacaldara | stephencurry30 | kingjames |
| 5 | wojciech.szczesny1 | willsmith | davidbeckham | willsmith | therock | thiagosilva | therock |
| 6 | 433 | natgeo | paulomeixedo | neymarjr | davidbeckham | davidbeckham | leomessi |
| 7 | theo3hernandez | therock | sergioramos | champagnepapi | blessedmma | gianlucavacchi | willsmith |
| 8 | filipefangueiro | nike | alvaromorata | kingjames | kingjames | leomessi | thenotoriousmma |
| 9 | champagnepapi | thenotoriousmma | wojciech.szczesny1 | kevinhart4real | neymarjr | ronaldo | 433 |
| 10 | official\_pepe | champagnepapi | blessedmma | davidbeckham | champagnepapi | djokernole | badgalriri |
| 11 | rickyregufe | badgalriri | kingjames | beautifuldestinations | badgalriri | neymarjr | nike |
| 12 | ufc | instagram | marcelotwelve | instagram | nike | zidane | natgeo |
| 13 | olivier\_rousteing | beyonce | rickyregufe | therock | leomessi | 433 | sergioramos |
| 14 | alvaromorata | kevinhart4real | thenotoriousmma | badgalriri | thenotoriousmma | rogerfederer | mattiacaldara |
| 15 | davidbeckham | leomessi | skysport | jlo | beyonce | stevengerrard | iamzlatanibrahimovic |

Table 6. Spearman Correlation.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | betweenness | degree | closeness | kpath | pagerank | k\_hop | k\_shell | Running Time |
| betweenness | 1 | 0.846549 | 0.583569 | 0.658302 | 0.680519 | 0.69079 | 0.796353 | 33.02118993 |
| degree | 0.846549 | 1 | 0.605392 | 0.750438 | 0.741967 | 0.832892 | 0.960802 | 0.000976086 |
| closeness | 0.583569 | 0.605392 | 1 | 0.687793 | 0.84742 | 0.761688 | 0.587425 | 38.12242889 |
| kpath | 0.658302 | 0.750438 | 0.687793 | 1 | 0.920179 | 0.930176 | 0.676465 | 5.928251982 |
| pagerank | 0.680519 | 0.741967 | 0.84742 | 0.920179 | 1 | 0.930126 | 0.698689 | 1.155621052 |
| k\_hop | 0.69079 | 0.832892 | 0.761688 | 0.930176 | 0.930126 | 1 | 0.800792 | 0.306159019 |
| k\_shell | 0.796353 | 0.960802 | 0.587425 | 0.676465 | 0.698689 | 0.800792 | 1 | 0.264872074 |

From Table 5, @cristiano is always in the top of ranking no matter the algorithm. This is reasonable, because @cristiano is the first user that the scraping project started to scrape, and all other scraped users are according to the followings of @cristiano. In this network, @cristiano has a strong link with other nodes.

Unlike other measures, K-Shell decomposition algorithm is just a coarse-grained solution to divide a network into several layers, but all node’s K-Shell index in same layer are the same. As a result, it is hard to rank user influence in one layer. In Table 5, all nodes’ K-Shell index es are same. Therefore, K-Shell decomposition algorithm does not apply to rank subtly user influence.

From Table 6, it is clear that the K-Path (5.93 seconds) is faster than Betweenness Centrality (33.02 seconds) and the spearman correlation between Betweenness and K-Path is higher than 0.5, that means the two metrics are relevant. Table 6 can also present that K-Hop needs 0.3061 seconds to compute, which is shorter than the computation time of Betweenness and Closeness Centrality. However, despite that, the running time is a little longer than expected as the author of K-Hop centrality indicated that “*K-hop centrality is more effective …, while it has lower computational complexity than betweenness centrality, closeness centrality, Kats centrality and k-shell*.” But from the result, K-Hop is not faster than k-shell. There are possible reasons for this: (1) the running time of K-Hop really depends on the value of K (set to 3 in this experiment). The higher value of k indicates the more network topology structure it has considered, but meanwhile, the longer time it needs. When k is set to 2, the running time is much shorter, about 0.08 seconds. (2) The K-Hop centrality was implemented in this experiment without great performance optimization. (3) The network is not big enough. The advantage of low computational complexity in K-Hop might be reflected in bigger networks.

## **The result of Topic-sensitive Algorithms**

### **TwitterRank**

Table 7 shows the Topic Distillation by LDA topic model based on the preprocessed user posts. By using LDA topic model, all post can be divided into 10 topics, and each topic has its topic words. The arrangement of the topic word in a certain topic is based on the the importance of the word in this topic.

Table 7. Topic Distillation by LDA

|  |  |  |  |
| --- | --- | --- | --- |
| **Topic** | **Topic Words** | **Topic** | **Topic Words** |
| 1 | follow, denim, shop, code, | 6 | link, photo, read, show, |
| 2 | watch, fashion, black, nike, | 7 | time, life, day, work, |
| 3 | photo, hawaii, travel, follow, | 8 | fashion, makeup, style, dia, |
| 4 | para, uma, porto, dia, | 9 | love, day, good, time, |
| 5 | con, milan, italia, madrid, | 10 | design, home, interior, decor, |

Table 8. Top 15 of Topical Ranking by TwitterRank

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| rank | topic1 | topic2 | topic3 | topic4 | topic5 |
| 1 | cristiano | hypebeast | natgeo | fcporto | skysport |
| 2 | natgeo | cristiano | beautifuldestinations | cristiano | calciatoribrutti |
| 3 | hypebeast | natgeo | cristiano | neymarjr | azzurri |
| 4 | instagram | snoopdogg | blessedmma | jbalvin | cristiano |
| 5 | neymarjr | houseofhighlights | snoopdogg | laliga | juventus |
| 6 | fcbarcelona | ufc | annalewandowskahpba | natgeo | gliautogol |
| 7 | juventus | bleacherreport | hypebeast | maluma | calciatoriignoranti |
| 8 | houseofhighlights | thebillionairesclub | ufc | ikercasillas | natgeo |
| 9 | bleacherreport | instagram | hawaiisbestkitchens | official\_pepe | fede\_nargi |
| 10 | jbalvin | therock | 9gag | fcbarcelona | papugomez\_official |
| 11 | 9gag | kyliejenner | besiktas | danialves | neymarjr |
| 12 | therock | kevinhart4real | therock | marcelotwelve | acmilan |
| 13 | 433 | blessedmma | bleacherreport | realsociedad | 433 |
| 14 | beautifuldestinations | 9gag | houseofhighlights | 9gag | mattiacaldara |
| 15 | snoopdogg | juventus | beautifulhotels | sergioramos | goldtattoo7 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| rank | topic6 | topic7 | topic8 | topic9 | topic10 |
| 1 | natgeo | natgeo | natgeo | snoopdogg | ashleytstark |
| 2 | hypebeast | snoopdogg | cristiano | ufc | dezeen |
| 3 | vogueitalia | ufc | hypebeast | cristiano | natgeo |
| 4 | cristiano | cristiano | vogueitalia | natgeo | cristiano |
| 5 | snoopdogg | therock | snoopdogg | bleacherreport | decor67 |
| 6 | vogueparis | hypebeast | anna\_dello\_russo | kevinhart4real | brabbu |
| 7 | riccardotisci17 | houseofhighlights | houseofhighlights | blessedmma | wallpapermag |
| 8 | anna\_dello\_russo | bleacherreport | riccardotisci17 | therock | dactylion\_design |
| 9 | i\_d | 9gag | i\_d | houseofhighlights | divine\_design\_decor |
| 10 | beautifuldestinations | kevinhart4real | neymarjr | neymarjr | instagram |
| 11 | badgalriri | blessedmma | beautifuldestinations | 9gag | hypebeast |
| 12 | britishvogue | champagnepapi | suzymenkesvogue | champagnepapi | beautifuldestinations |
| 13 | suzymenkesvogue | instagram | bleacherreport | hypebeast | thebillionairesclub |
| 14 | instagram | joerogan | badgalriri | 433 | onlyforluxury |
| 15 | voguehommes | neymarjr | juventus | juventus | architectanddesign |

A screenshot of a cell phone

Description automatically generated From Table 8, it can be seen that no matter how topic changes, @cristiano is always still in the top. This is because TwitterRank is indeed based on PageRank, and PageRank cares about user “following” relationship. It can also be found that @snoopdog, @hypebeast and @ bleacherreport appear in the top15 of every topical ranking frequently despite it does not appear so frequently in general ranking. This phenomenon presents the importance of another factor in TwitterRank: the number of posts the user posted in Instgram. Figure 6 explains that those users are not very important in the network according to centralities in the basic experiment, but they all appear in topical ranking in TwitterRank frequently because they post lots of posts with content. That means a user who likes to post posts in Instagram is more likely to be placed on high position in topical ranking by TwitterRank.

Figure 6. Ranking by the number of posts

### **MentionRank**

Table 9. Top 15 of Topical by MentionRank

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Home | | Fashion | | Sport |
| rank | **MentionRank** | **TwitterRank Topic10** | **MentionRank** | **TwitterRank Topic8** | **MentionRank** |
| 1 | decor67 | ashleytstark | pg\_dmcasting | natgeo | nike |
| 2 | atelier\_di\_spera | dezeen | vogueportugal | cristiano | briandecosta |
| 3 | ashleytstark | natgeo | officialwatches | hypebeast | marcusfilly |
| 4 | brabbu | cristiano | atelier\_di\_spera | vogueitalia | sportingclubedeportugal |
| 5 | livingedgeinteriors | decor67 | modamasculina | snoopdogg | pierreseliteperformance |
| 6 | lux.interiors | brabbu | nelly\_goncalves | anna\_dello\_russo | nikefootball |
| 7 | onlyforluxury | wallpapermag | gentbelike | houseofhighlights | gr.sports |
| 8 | les\_belles\_collections\_ | dactylion\_design | portugalfashion | riccardotisci17 | theislandwolfjv |
| 9 | archiraffa | divine\_design\_decor | off\_\_\_\_white | i\_d | miketyson |
| 10 | luxurystone | instagram | gentrygarb | neymarjr | performsmc |
| 11 | divine\_design\_decor | hypebeast | fashion\_mags | beautifuldestinations | skysport |
| 12 | ferrisrafauli | beautifuldestinations | cristanini\_viaggi | suzymenkesvogue | joerogan |
| 13 | marvelous\_listings | thebillionairesclub | blazmedclothing | bleacherreport | arsenal |
| 14 | uniqueluxurystyle | onlyforluxury | sneakerjeans | badgalriri | ufc |
| 15 | thegreycollective | architectanddesign | lux.interiors | juventus | cr7crunchfitness |

It is a very interesting result. First, from the topical ranking by MentionRank, it is clear that user influences have much difference upon different topics and the top15 users on its field really are really professional to their specific topics. For example, in the home page of @decor67, the top 1 user under “Home” topic, there are all pictures about different styles of homes. One other thing to note is that there are two same users in two different topics (Home and Fashion) by MentionRank: @ atelier\_di\_spera and @ lux.interiors. by observing their posts, in the strict sense of the term, those two users both should be classified to “Home” topic, but actually #fashion hashtags can also be used in Home-related posts because “home” is a kind of “fashion”.

Because TwitterRank uses LDA topic model to extract topic information, and LDA generates 10 topics based on posts without “topic title”, so it is not easy to compare topical ranking on same topic by TwitterRank and MentionRank. According to topic words created by LDA, *topic10* can be viewed as “Home” topic and *topic8* can be viewed as “Fashion” topic tentatively. Unfortunately, because of the limitation of LDA, there no sport-related topic created. In the ranking of “Home” topic by TwitterRank, there are 9 users really about “home”, and other 6 users either have strong “following” relationships or have posted lots of posts in Instagram. Similarly, in the ranking of “Fashion” topic by TwitterRank, there are just 6 “professional” users in Fashion.

## **The result of visualization:**

This visualization of this project is a web-based application, including functionalities of scraping, analyzing and visualizing.

On the scraping page, there is an input box for user to input a number to scrape the number of Instagram users. The default number is 10. And after clicking the “Go” button, the scraping project begins to scrape. A dialog will be shown to the site to reminder users of the loading process. The length of scraping times really depends on the inputted username.

A screenshot of a person

Description automatically generated A screenshot of a cell phone

Description automatically generated

Figure 7 Figure 8

After finishing the scraping word, all the 10 users’ following relations will be listed in the table. But because the graph just focuses on the 10 users, so other irrelevant users should be removed before analyzing and visualizing.

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

Filter

Figure 9

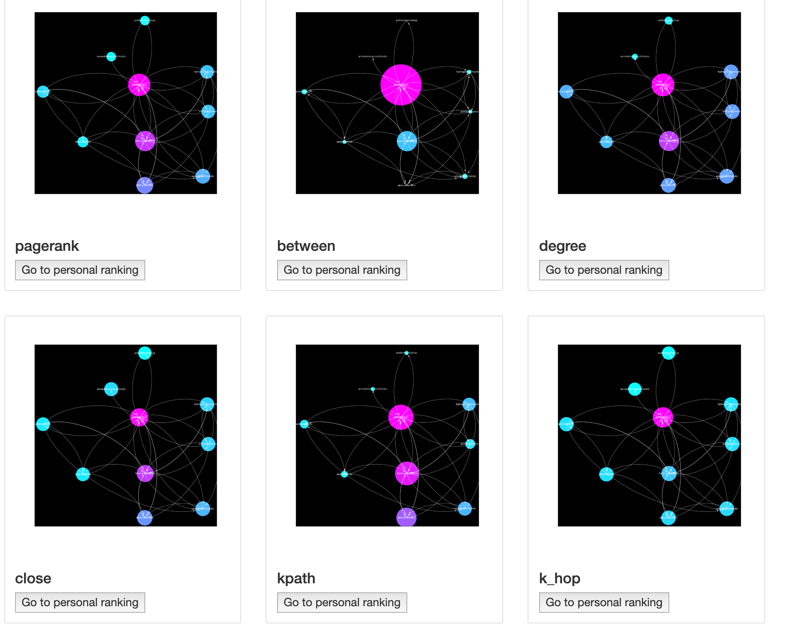
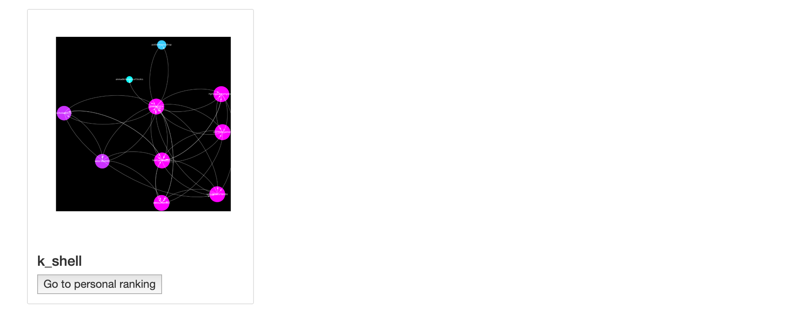


Figure 10

After filtering, user can click “*analyzing and visualize*” button to go to the next page to see the result of the 10 users influential ranking by different influential measures. In the result page, there are seven pictures corresponding to the results of the seven algorithms above. Nodes in different pictures are all in the same positions which are calculated by cluster analysis, but they are different in colors and sizes. The more red and bigger in one visualization, the more important the node is by this measure.

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

Figure 11

Figure 12

Then users can go to next page to see the ranking list clearly by clicking the button in corresponding visualization. In this “*ranking list*” page, users can see the what the exact score every node has by the measure and go to a personal page to see more information about this Instagramer (Figure 11).

In the personal page, users also can obtain the wordcloud of the Instagramer by scraping. The greatest number of the posts which can be scrape in this system is set to 200, so it will not take a long time to scrape (Figure 12).

# **Evaluation**

## **Graph-Based Approaches**

In order to evaluate Graph-Based Approaches, IC model (Independent Cascade Model) was implemented as Influence Diffusion Model. To ensure the accuracy of the experimental result, 100 simulation experiments has been made for each result of Top-n ranking list by different measures, and the averaged result is used to draw the figure of user influence diffusion. In every evaluative experiment, an activation probability matrix *P* would be created randomly, *P*(*u,v*) indicates the probability that node *u* activates node *v*. For this project, it can also mean the probability that node *u* influences node *v*. In the implementation of the IC model, a list structure was applied to show a set of influential nodes *S* (could be Top-n users by one measure). Traversal depth *M* was set to present the time of influence diffusion. We assume that after M depths information propagation, information could not spread anymore, and other nodes could not be influenced by the source node. Therefore, after a certain node *i* has walked M depths(steps) in the network, the simulation is over.

As a result, the evaluation criterion is based on the activated/influenced ratio that the sphere of influence by the seed set after simulation experiments are over. The greater influenced ratio value, the higher performance of the algorithm, that means the higher quality of the seed set chosen by the algorithm.

The influenced ratio is expected to be no longer changeable greatly after *M* depth. So, before the simulation experiment begins, traversal depth *M* should be initialized appropriately. The following figure represents that the node influenced ratio expends as time/step goes by different measures.

A close up of text on a white background

Description automatically generated

Top-200 influential users

Top-100 influential users

Top-20 influential users

Figure 13. Node Density depending on time and top-n nodes

From the figure, it can be clearly seen that with an increasing number of seeds (influential nodes), the speed of the growing of the influenced ratio is up. For most of measures, after 3 depth/steps, Top 20 and Top 100 influential users can activate/influence 85% nodes in the network, and the ratio will not be increased obviously. For Top 200 users, only after 2 depth/steps, the growth curve of influenced ratio approaches to a flat. This result is reasonable, so, *M* could be set to 4 in the simulation experiments.

In order to make the result clear, the influenced ratio is presented as bar chart and the Y-axis range is between 0.84 and 0.87.

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

Figure 14. Node Density by Measures

In Figure 14, there are 3 bar charts show the influenced ratio of Top20/Top100/Top200 influential users respectively by different measures.

In the “Top20” bar chart, it is clear to seen that the result of Betweenness, Degree and K-Shell Centralities are similar, and all lower than that of other measures. That means that the quality of the Top20 influential users measured by the three Centralities is lower than that by others. The performance of PageRank is higher than other measures by the Influence Diffusion Model. Other three centralities, Closeness, K-Path, and K-Hop Centralities, have similar performance.

In the “Top100” and “Top200” bar charts, with the increasing number of Top influential users, it is obvious that almost all measures expend the influence except Betweenness Centrality and K-Shell. It can be inferred that their ability from Top20 to spread is undifferentiated with that from the Top100 even Top200, and the spreading ability is still low. The result is expected: (1) Betweenness Centrality is to find user acting as “bridge” in a network. But in the Instagram network, the relationships among most users are close, and there are no separated clusters, so, it would not appear the phenomenon that a node has very privileged power connecting with two or more separated groups. (2) On the other hand, K-Shell is a kind of decomposition, not a ranking measure. Therefore, K-Shell cannot rank user influence in a same layer. K-Path and PageRank have almost the same influence ratio in the both bar charts. This conforms the Spearman Correlation between the two measures (0.92). Even the running time of K-Path (5.93 seconds) is longer than that of PageRank (1.16), K-Path is much more flexible and dynamic, because PageRank is a global measure, every time the network has any little change, all nodes’ PageRank score should be recalculated again. But K-Path is a k-local measure, so only relevant nodes’ K-Path score should be calculated again (K-Hop has the same feature). In this sense, especially in large-scale networks, K-Path is much more efficient. K-Hop is a generalization of Degree Centrality, and it is clear that the performance of K-Hop is always much higher than Degree Centrality, but always a little lower than K-Path and PageRank. Nevertheless, K-Hop takes much shorter time (0.31 seconds). In aggregate, K-Hop is the highest cost-effective: (1) K-Hop takes short time. (2) The performance of K-Hop is not bad. It is predictable that Closeness Centrality always has highest performance, because the nature of Closeness Centrality very accord with the idea of IC model. Simply, Closeness Centrality is to put nodes which has shorter distances to all other nodes in the network on a higher ranking. So, nodes with higher Closeness score is more like a gossiper, who can disseminate information to neighbors and neighbors’ neighbors. However, because Closeness Centrality is a global measure, so the running time in the network is very long (38.12 seconds).

## **Topic-Sensitive Algorithms**

According to the result of the ranking of TwitterRank and Mentionrank, there is a tentative conclusion that (1) MentionRank is more topic-sensitive, because Top15 users by Mentionrank are all highly related with the given topic, but whether the Top15 users have higher influence on the topic than that by Twitterrank, that is still not determined. (2) TwitterRank is based on PageRank, so the users with high TwitterRank are indeed influential, but it is hard to say whether the users have also high influence in this given topic. In order to figure out the puzzle, IC model still can be implemented in the contrast test.

In the topical evaluation, most steps are same with the former one except for the initialization of the activation probability matrix *P*. In this evaluation, *P* was not created randomly but according to the ranking result of MentionRank. The concrete initialization rules as follows:

Table 10. The Initialization of ***P*(*u,v*)**

|  |  |
| --- | --- |
| The Topical Ranking of *u* by MentionRank | *P*(*u,v*) |
| Top 20 | 0.9 |
| Top20-Top100 | 0.7 |
| Top100- Top200 | 0.6 |
| Other | 0.5 |

The initialization for *P* is a hypothesis. The reason for the initialization is based on the idea of that a topic-related professional user is more likely to influence others upon this topic. So based on the hypothesis, Figure 12 shows the result of the influenced ratio of Top20/Top100/Top200 influential users respectively by MentionRank and TwitterRank on the topic “Home” and “Fashion”.

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

Figure 12. Node Density by Topic-sensitive Measures

### **“Home” Topic:**

From the analysis of Result Section above, there are 9 “home”-related users among Top15 by TwitterRank, so in fact, we can infer that the Topic 20 by TwitterRank contains a certain number of “home”-related users as well as celebrities. In the “Top20” figure, the influenced ratio by TwitterRank is higher than that by MentionRank on “Home” topic. But in the “Top100” figure, the influenced ratio by MentionRank is higher. In the Top100 users TwitterRank, there are more celebrities, but it seems that normal users are more likely to be influenced by “home”-related users measures by MentionRank. Therefore, it can be summarized that on “Home” topic, “home”-related users (like interior designer, home market or architect) have more influence.

### **“Fashion” Topic:**

From Figure 12, it is easy to find that the influenced ratios by MentionRank are always a little lower than that by TwitterRank based on 3 different Top-n figures. It suggests that the quality of Top-n seeds by MentionRank is not good as that by TwitterRank. Accordingly, it can be inferred that on “Fashion” Topic the influence of professional users (like model, photographer or fashion magazine editor) is lower than that of celebrities who have lots of high-quality fans (like singer, actor, or football player).

# **Conclusions**

**References**

1. [↑](#footnote-ref-1)