INTERNATIONAL JOURNAL OF RESEARCH AND DEVELOPMENT IN ENGINEERING SCIENCES
e-ISSN: 2582 - 4201 Volume: 2 | Issue: 1 | Year: 2020

PP Rank: Economically Selecting Initial Users for Influence Maximization in Social Networks

K.KANTHARAJU

Assistant Professor and Head of the Department
Department of Computer Science and Engineering,
GDMM College of Engineering, Nandigama, Krishna Dist.,
Jawaharlal Nehru Technological University Kakinada, Andhra Pradesh, India
e-mail: kantharaju8@gmail.com

Abstract: This paper focuses on seeking a new heuristic scheme for an influence maximization problem in social networks: how to economically select a subset of individuals (so-called seeds) to trigger a large cascade of further adoptions of a new behavior based on a contagion process. Most existing works on selection of seeds assumed that the constant number k seeds could be selected, irrespective of the intrinsic property of each individual's different susceptibility of being influenced (e.g., it may be costly to persuade some seeds to adopt a new behavior). In this paper, a price-performance-ratio inspired heuristic scheme, PP Rank, is proposed, which investigates how to economically select seeds within a given budget and meanwhile try to maximize the diffusion process. Our paper's contributions are threefold. First, we explicitly characterize each user with two distinct factors: the susceptibility of being influenced (SI) and influential power (IP) representing the ability to actively influence others and formulate users' SIs and IPs according to their social relations, and then, a convex price-demand curve-based model is utilized to properly convert each user's SI into persuasion cost (PC) representing the cost used to successfully make the individual adopt a new behavior.

Furthermore, a novel cost-effective selection scheme is proposed, which adopts both the price performance ratio (PC-IP ratio) and user's IP as an integrated selection criterion and meanwhile explicitly takes into account the overlapping effect; finally, simulations using both artificially generated and real-trace network data illustrate that, under the same budgets, PPRank can achieve larger diffusion range than other heuristic and brute-force greedy schemes without taking users' persuasion costs into account.

Keywords: Social Networks, Greedy Algorithm, Influential power, PP Rank

1. INTRODUCTION

Humans in social networks behave in a viral fashion, having a natural inclination to spread information (or behavior), specifically in which users coordinate their decisions and form conventions by being influenced by the behaviors of their friends or neighbors, e.g., whether to adopt a new behavior or not. Basically, the influence diffusion pattern in social networks can be composed of diffusion models and diffusion process. Diffusion models describe how the individual determines whether to adopt behavior (e.g., spread information) or not, including general threshold model and cascade model, etc.; the diffusion process characterizes the unfolding of the dynamics of behaviors adopted by the whole population in a network (through a contagion process like epidemic).

Recently, study of influence propagation in social networks has gained tremendous attentions [1], [2]. In particular, we focus on the influence maximization problem first stated by Domingos and Richardson [3]: Given a specific diffusion model, if a subset of individuals could be convinced to adopt a new product and the goal is to trigger a large cascade of further adoptions, which set of individuals should be targeted in order to achieve a maximized influence? It was shown that finding the influential set of initial nodes (so-called seeds) is an NP-hard problem, and only for the sub modular function of the diffusion model, a simple greedy algorithm (choosing the nodes with maximal marginal gain) could approximate the optimal solution by a $(1 - 1/e)$, i.e., within 63% of optimal [4].

However, the simple greedy-based approach has a heavy computation load. Specifically, greedy-based algorithms calculate the influence power precisely by enumeration. The more rounds the enumeration takes, the more accurate the result will get. However, when the network size increases, the computational time will increase dramatically, which prevents the greedy based algorithm to become a feasible solution for the influence maximization problem in real world.

Research community mainly solves the aforementioned efficiency problem from two directions: The first is to improve those greedy algorithms to further reduce their running time, and the second is to propose new heuristics that utilize some structural properties of social networks to directly select seeds at one time without running tremendous enumerations to infer each user's influence power.

Reference [5] presented an optimized greed algorithm, which is referred to as the "cost-effective lazy forward (CELF)" scheme. The CELF optimization uses the sub modular property of the influence maximization objective to reduce the number of evaluations on the influence spread of nodes. Their experimental results demonstrate that, in comparison with simple greedy-based approach, CELF optimization could achieve as much as 700 times speedup in selecting seeds. The CELF strategy can greatly reduce the number of evaluations on the influence spread of nodes. This strategy was further improved to a CELF++ strategy [6], which simultaneously calculates the influence spread for two successive iterations of a greedy algorithm. IEEE Systems Journal, Considering that the fundamental step of the greedy-based algorithm is to pick a node in each iteration from the remaining nodes, with attempting to make the maximum marginal contribution to the process of spread of information, SPIN (ShaPley value-based Influential Nodes)was proposed to compute the marginal contributions using the concept of Shapley (a well-known solution concept in cooperative game theory).

Specifically, the Shapley value of a coalitional game provides the marginal contribution of an individual player to the overall value that can be achieved by the grand coalition of all of the players. Reference [8] provided an approach to selecting a finite number of influential social sensors, based on graph sampling, by which the search space can be dramatically reduced.

However, considering that those improved algorithms are essentially greedy based, their running times are still long. A possible alternative is to use heuristics. In sociology literature, degree and other centrality-based heuristics are commonly used to estimate the influences of nodes in social networks. However, if all seeds are chosen solely based on the measurement of centrality, it is shown that the resulting scheme only outperforms random selection, due to overlapping effect [9]. By overlapping effect, it means that a given group of connected nodes may have a high degree, but if their adjacent nodes are overlapped, then behavior may not be widely propagated into the rest of the social networks.

2. PRELIMINARY INVESTIGATION

The first and foremost strategy for development of a project starts from the thought of designing a mail enabled platform for a small firm in which it is easy and convenient of sending and receiving messages, there is a search engine ,address book and also including some entertaining games. When it is approved by the organization and our project guide the first activity, ie. Preliminary investigation begins. The activity has three parts:

- Request Clarification
- Feasibility Study
- Request Approval

Request Clarification: After the approval of the request to the organization and project guide, with an investigation being considered, the project request must be examined to determine precisely what the system requires. Here our project is basically meant for users within the company whose systems can be interconnected by the Local Area Network (LAN). In today's busy schedule man need everything should be provided in a readymade manner. So taking into consideration of the vastly use of the net in day to day life, the corresponding development of the portal came into existence.

Feasibility Analysis: An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are

- Operational Feasibility
- Economic Feasibility
- Technical Feasibility

Operational Feasibility: Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

Economic Feasibility: Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at anytime. The Virtual Private Network is to be developed using the existing resources of the organization. So the project is economically feasible.

Technical Feasibility: According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Java Server Pages, JavaScript, HTML, SQL server and Web Logic Server are used to develop the system. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

Request Approval: Not all request projects are desirable or feasible. Some organization receives so many project requests from client users that only few of them are pursued. However, those projects that are both feasible and desirable should be put into schedule. After a project request is approved, it cost, priority, completion time and personnel requirement is estimated and used to determine where to add it to any project list. Truly speaking, the approval of those above factors, development works can be launched.

3. EXISTING SYSTEM

In this section, we introduce the iterative ranking algorithm, which takes an iterative process to measure the vitality of users within a social network Consider the mutual influence

between users while computing the vitality measurements and propose the second ranking algorithm, which computes user vitality in an iterative way. Other than user vitality ranking for many time instructions with social Network. In this paper, we propose aunique perspective to achieve this goal, which is quantifying user vitality by analyzing the dynamic interactions among users on social networks. Examples of social network include but are not limited to social networks in microblog sites and academicals collaboration networks.

Intuitively, if a user has many interactions with his friends within a time period and most of his friends do not have many interactions with their friends simultaneously, it is very likely that this user has high vitality. Based on this idea, we develop quantitative measurements for user vitality and propose our first algorithm for ranking users based vitality. Also we further consider the mutual influence between users while computing the vitality measurements and propose the second ranking algorithm, which computes user vitality in an iterative way.

JDBC versus ODBC and other APIs

At this point, Microsoft's ODBC (Open Database Connectivity) API is that probably the most widely used programming interface for accessing relational databases. It offers the ability to connect to almost all databases on almost all platforms. So why not just use ODBC from Java? The answer is that you can use ODBC from Java, but this is best done with the help of JDBC in the form of the JDBC-ODBC Bridge, which we will cover shortly. The question now becomes "Why do you need JDBC?" There are several answers to this question:

1. ODBC is not appropriate for direct use from Java because it uses a C interface. Calls from Java to native C code have a number of drawbacks in the security, implementation, robustness, and automatic portability of applications.

2. A literal translation of the ODBC C API into a Java API would not be desirable. For example, Java has no pointers, and ODBC makes copious use of them, including the notoriously error-prone generic pointer "void *". You can think of JDBC as ODBC translated into an object-oriented interface that is natural for Java programmers.

3. ODBC is hard to learn. It mixes simple and advanced features together, and it has complex options even for simple queries. JDBC, on the other hand, was designed to keep simple things simple while allowing more advanced capabilities where required.

4. A Java API like JDBC is needed in order to enable a "pure Java" solution. When ODBC is used, the ODBC driver manager and drivers must be manually installed on every client machine. When the JDBC driver is written completely in Java, however, JDBC code is automatically installable, portable, and secure on all Java platforms from network computers to mainframes.

4. PROPOSED SYSTEM

In this section, in social networks behave in a viral fashion, having a natural inclination to spread information (or behavior), specifically in which users coordinate their decisions and form conventions by being influenced by the behaviors of their friends or neighbors a new behavior has occurred, then we focus on its spreading in social networks. Each user, classified as either an "active" or a "potential" consumer, is represented by a node in the social network structure. An active consumer is a user who has already adopted the behavior and so can influence her neighbors in favor of obtaining it.

A potential consumer is an individual who does not adopt the behavior yet but is susceptible of obtaining it if exposed to someone who does. Social networks can be composed of diffusion models and diffusion process. Diffusion models describe how the individual determines whether to adopt behavior (e.g., spread information) or not, including general threshold model and cascade model, etc. This paper we may be share to products details to neighbors and view more instruction user .all details share to others through this projects and can find out for influence group (social networks).you sharing for neighbors details to only include group.

Advantages: This project used to add advertisement and share user to this add for neighbors.

ALOGITHMS:

Greedy algorithm

A greedy algorithm is an algorithmic paradigm that follows the problem solving heuristic of making the locally optimal choice at each stage[1] with the hope of finding a global optimum. In many problems, a greedy strategy does not in general produce an optimal solution, but nonetheless a greedy heuristic may yield locally optimal solutions that approximate a global optimal solution

Greedy algorithm (choosing the nodes with maximal marginal gain) could approximate the optimal solution by a $(1 - 1/e)$, i.e., within 63% of optimal However, the simple greedy-based approach has a heavy computation load. Specifically, greedy-based algorithms calculate the influence power precisely by enumeration. The more rounds the enumeration takes, the more accurate the result will get. However, when the network size increases, the computational time will increase dramatically, which prevents the greedy based algorithm to become a feasible solution for the influence maximization problem in real world.

Heuristic algorithm

The term heuristic is used for algorithms which find solutions among all possible ones ,but they do not guarantee that the best will be found, therefore they may be considered as

approximately and not accurate algorithms. These algorithms, usually find a solution close to the best one and they find it fast and easily. Sometimes these algorithms can be accurate, that is they actually find the best solution, but the algorithm is still called heuristic until this best solution is proven to be the best. The method used from a heuristic algorithm is one of the known methods, such as greediness, but in order to be easy and fast the algorithm ignores or even suppresses some of the problem's demands.

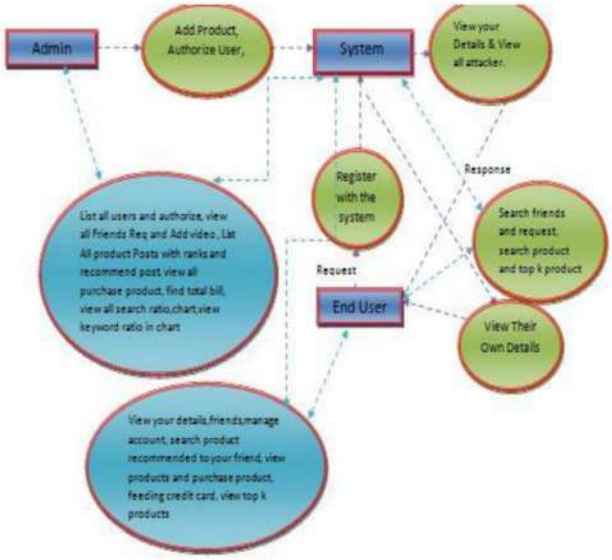


Figure.1: Data Flow Diagram

System Configuration:

Hardware Requirements:

Hardware	-	Pentium
Speed	-	1.1 GHz
RAM	-	1GB
Hard Disk	-	20 GB
Floppy Drive	-	1.44 MB
Key Board	-	Standard Windows Keyboard
Mouse	-	Two or Three Button Mouse
Monitor	-	SVGA

Software Requirements:

Operating System	: Windows
Technology	: Java and J2EE
Web Technologies	: Html, JavaScript, CSS
IDE	: My Eclipse(mars 2.0)
Web Server	: Tomcat 8.0
Database	: My SQL
Java Version	: jdk 1.8

Validation Checking

Validation checks are performed on the following fields.

Text Field: The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

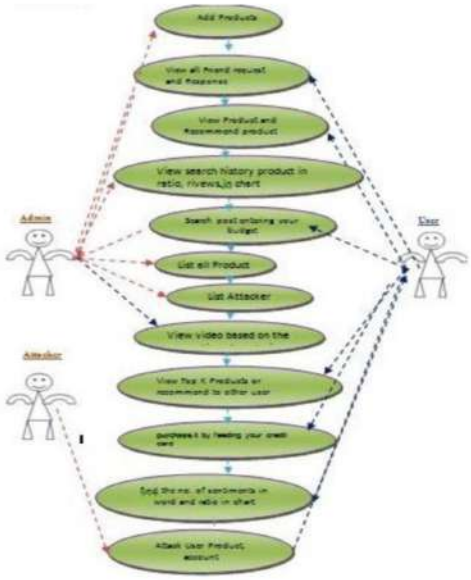


Figure.2: Use case Diagram

Numeric Field: The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested. A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

Preparation of Test Data: Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

Using Live Test Data: Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have they entered themselves. It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is

realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system. This bias toward typical values then does not provide a true systems test and in fact ignores the cases most likely to cause system failure.

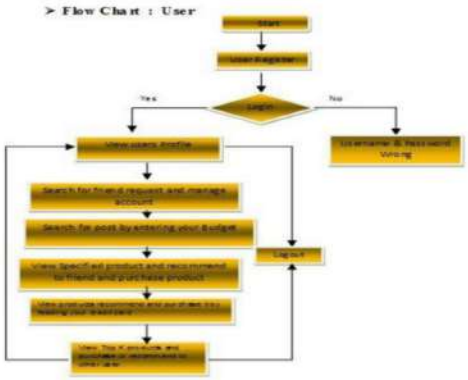
Using Artificial Test Data: Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program. The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

USER TRAINING: Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy.

MAINTAINENCE: This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user's requirements during the process of system development. Depending on the requirements, this system has been developed to satisfy the needs to the largest possible extent. With development in technology, it may be possible to add many more features based on the requirements in future. The coding and designing is simple and easy to understand which will make maintenance easier.

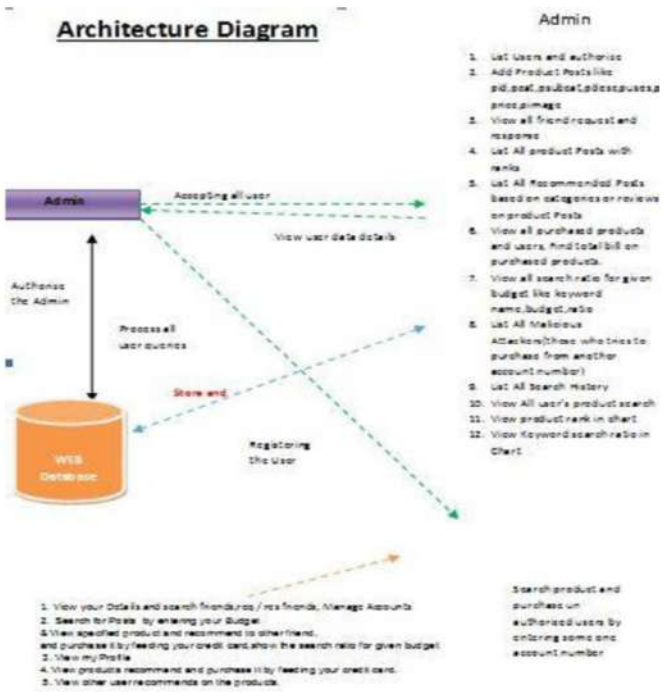
TESTING STRATEGY: A strategy for system testing integrates system test cases and design techniques into a well planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation.

A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements. Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing.



SYSTEM TESTING

SYSTEM TESTING: Software once validated must be combined with other system elements (e.g. Hardware, people, and database). System testing verifies that all the elements are proper and that overall system function performance is achieved. It also tests to find discrepancies between the system and its original objective, current specifications and system documentation.



UNIT TESTING: In unit testing different are modules are tested against the specifications produced during the design for the modules. Unit testing is essential for verification of the code produced during the coding phase, and hence the goals to test the internal logic of the modules. Using the detailed design description as a guide, important Conrail paths are tested to uncover errors within the boundary of the modules.

This testing is carried out during the programming stage itself. In this type of testing step, each module was found to be working satisfactorily as regards to the expected output from the module.

In Due Course, latest technology advancements will be taken into consideration. As part of technical build-up many components of the networking system will be generic in nature so that future projects can either use or interact with this. The future holds a lot to offer to the development and refinement of this project.

IMPLEMENTATION

- **Admin:** In this module, the Admin has to login by using valid user name and password. After login successful he can perform some operations such as List Users and authorize and add Product Posts like pid,pcat,psubcat,pdesc,puses,pprice,pimage ,View all friend request and response ,List All product Posts with ranks and List All Recommended Posts based on categories or reviews on product Posts ,View all purchased products and users, Find total bill on purchased products, View all search ratio for given budget like keyword name,budget,ratio ,List All Malicious Attackers(those who tries to purchase from another account number),List All Search History ,View All user's product search and View product rank in chart, view Keyword search ratio in Chart
- **Friend Request & Response:** In this module, the admin can view all the friend requests and responses. Here all the requests and responses will be displayed with their tags such as Id, requested user photo, requested user name, user name request to, status and time & date. If the user accepts the request then the status will be changed to accepted or else the status will remains as waiting.
- **User1:** In this module, there are n numbers of users are present. User should register before performing any operations. Once user registers, their details will be stored to the database. After registration successful, he has to login by using authorized user name and password. Once Login is successful user can perform some operations like View your Details and search friends,req / res friends, Manage Accounts ,Search for Posts by entering your Budget & View specified product and recommend to other friend, and purchase it by feeding your credit card,show the search ratio for given budget and View my Profile ,View products recommend and purchase it by feeding your credit card,View other user recommends on the products. and View Top K products and purchase or recommend to other user.
- **Searching Users to make friends:** In this module, the user searches for users in Same Network and in the Networks and sends friend requests to them. The user can search for users in other Networks to make friends only if they have permission.
- **Attacker:** Search product and purchase un authorized users by entering some one account number

5. SYSTEM DESIGN AND DEVELOPMENT

INPUT DESIGN: Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design.

Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error is in the input are controlled by the input design. The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position where must be entered. The user is also provided within an option to select an appropriate input from various alternatives related to the field in certain cases. Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

OUTPUT DESIGN: The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself. A new user may be created by the administrator himself or a user can himself register as a new user but the task of assigning projects and validating a new user rests with the administrator only.

The application starts running when it is executed for the first time. The server has to be started and then the internet

explorer in used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act as the clients. The developed system is highly user friendly and can be easily understood by anyone using it even for the first time.

6. CONCLUTION

In this paper, given the underlying social network structure and influence model, our paper focuses on the interesting problem of maximizing influence propagation of new behavior in social networks. The literature has greatly studied the mentioned problem from two directions: the enhanced greedy algorithms and various heuristic schemes. However, all existing works ignore one key aspect of influence propagation that we usually experience in real social life: The cost used to persuade individuals to adopt a new behavior might vary highly (due to their different susceptibilities of being influenced). Thus, instead of being given a static number of initial seeds, the main motivation of our paper is to investigate how to economically select initial seeds within a given budget to maximize influence. To solve the aforementioned issue, this paper proposes a new heuristic algorithm, PPRank, based on the integration of price-performance ratio and influence power. the real social network data traces and artificially generated network data verify that, under same budgets, our proposal can achieve better performance than other heuristic and greedy-based schemes, in terms of diffusion range.

References

[1]. J. M. Kleinberg, "Cascading behavior in social and economic networks," in *Proc. 14th ACM Conf. EC*, 2013, pp. 1–4.

[2]. W. Chen, L. V. S. Lakshmanan, and C. Castillo, "Information and influence propagation in social networks," in *Synthesis Lectures on Data Management*. San Rafael, CA, USA: Morgan & Claypool Publishers, 2013.

[3]. P. Domingos and M. Richardson, "Mining the network value of customers," in *Proc. 7th KDD*, 2001, pp. 57–66.

[4]. D. Kempe, J. M. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proc. 9th KDD*, 2003, pp. 137–146.

[5]. J. Leskovec *et al.*, "Cost-effective outbreak detection in networks," in *Proc. of 13th KDD*, 2007, pp. 420–429.

[6]. A. Goyal, W. Lu, and L. V. S. Lakshmanan, "CELF++: Optimizing the greedy algorithm for influence maximization in social networks," in *Proc. of 20th WWW*, 2011, pp. 47–48.

[7]. R. Narayanam and Y. Narahari, "A Shapley value based approach to discover influential nodes in social networks," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 1, pp. 130–147, Jan. 2011.

[8]. J. Zhao, J. C. S. Lui, D. Towsley, X. Guan, and P. Wang, "Social sensor placement in large scale networks: a graph sampling perspective," *CoRR abs/1305.6489*, 2013.

[9]. B. Han and A. Srinivasan, "Your friends have more friends than you do: Identifying influential mobile users through random-walk sampling," in *Proc. 13th MOBIHOC*, 2012, pp. 5–14.

[10]. W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in *Proc. 15th KDD*, 2009, pp. 199–208.