Analysis of Main Expert-Finding Algorithms in Social Network in Order to Rank the Top Algorithms

Ahmad Agha Kardan*
Faculty of Computer Engineering, Amirkabir University of Technonlogy, Tehran, Iran
aakardan@aut.ac.ir
Behnam Bozorgi
Faculty of Computer Engineering, Amirkabir University of Technonlogy, Tehran, Iran
behnam.bozorgi71@gmail.com

Received: 23/Aug/2016 Revised: 08/May/2017 Accepted: 06/Aug/2017

Abstract

The ubiquity of the Internet and social networks have turned question and answer communities into an environment suitable for users to ask their questions about anything or to share their knowledge by providing answers to other users' questions. These communities designed for knowledge-sharing aim to improve user knowledge, making it imperative to have a mechanism that can evaluate users' knowledge level or in other words "to find experts". Experts are people who are highly talented in a specific field like technology, languages, cooking and etc, and we focused on specific fields in programming languages. There is a need for expert-finding algorithms in social networks or any other knowledge sharing environment like question and answer communities. So companies that looking for hiring programmers can easily find people suit their needs. There are various content analysis and link analysis methods for expert-finding in social networks. Experts can be identified by their behaviors in forums and their relationships in proportion to the questions and answers they share. Therefore, analyzing social networks can provide us with appropriate information. This paper aims to challenge four algorithms by applying them to our dataset and analyze the results in order to compare the algorithms. The algorithms suitable for expert finding has been found and ranked. Based on the results and tests, it is concluded that the Z-score algorithm has a better performance than others. The outcome of this article for enthusiasts is top algorithms for expert finding in question-answer communities. In this paper, first, we will start by introducing the problem of expert finding problem the related works will be presented then algorithms and dataset are introduced and finally results are illustrated.

Keywords: Expert-Finding; Social Network Analysis; Question and Answer Community; Stack Overflow.

1. Introduction

People face various obstacles during their lifetime; from various everyday problems to unique problems and from non-specialized to specialized problems. Everyone tries to solve these problems and will naturally look for those who are experts in that field so that they can solve the problem as soon as possible. Nowadays the first thing that most people do is to turn on their computer and search for an answer on the internet or ask their friends on social networks. But some who are more determined for finding accurate answers to their questions try various online communities where they are able to open a new discussion and ask their questions there. Question and answer (Q&A) communities are particularly suitable for people to share their questions with others and look for consultation from others. But oftentimes there are various, sometimes even contradictory, answers to a question provided by other users. So finding a correct answer would be very challenging. One of the solutions to the problem is to find the experts on these communities and only trust their words. In recent years there has been a great deal of interest in finding a solution for this need. Expert finding is process of identifying users who have the highest level of expertise in specific field of knowledge.

Q&A communities can fit two categories: the specialized and non-specialized. Specialized question and answer communities are those that focus on a particular field. For instance specialized Java communities are designed for questions and answers regarding Java programming. Yahoo Answer is an example of non-specialized question and answer communities, which cover a large number of topics. The objective of this paper is to challenge four algorithms and compare their performance in finding number of experts in specific field. Each algorithm focuses on specific aspects to find experts and that makes them unique to find experts in each field in this paper. But one thing in common in these algorithms is the fact that they try to find most important node in the network their unique way which make them suitable for finding experts in social media networks and These algorithms are the most extendable and Improvable algorithms available in expert-finding. The novelty of this research is to compare 6 main algorithms (including extended ones like directed-Indegree and Z-number and Z-degree) on a large-scale dataset including more than 1 million Q&A data which was hard to pre-process. This large amount of data to be processed makes the results real and trustable for future work of enthusiasts.

* Corresponding Author

This paper will be presented as described below:

In section 2 we will review the related literature. An introduction for the dataset will be in section 3. Section 4 describes expert finding algorithms while the analysis of the algorithms will be presented in section 5. Section 6 consists of the evaluation process of the algorithms and the conclusions will be presented in section 7. Future work will be presented in section 8.

2. Related Work

The rapid expansion of technology in recent years has resulted in a vast amount of information being published on the internet that can be useful in a variety of ways. One such use is investigated in [1] where they tried to find experts to facilitate knowledge-sharing. In this work they first discuss various expert-finding methods. There are two types of expert-finding, finding experts through analyzing user interaction, e.g. HITS-based algorithms and PageRank algorithm, and content analysis based methods. The authors of this paper state that the link analysis method is more successful than content-based method, but they did express some objections regarding this approach. They concluded that the algorithms like HITS always try to increase user scores. Of course this is only a problem when the dataset does not inherently include negative scoring. Therefore, users' links in their communicational network can be either negative or positive, an example of such links between two users can be seen in figure. 1. op is opinion of user n_j and r is relation of user n_i with user n_j where n_i gives alink point to specific opinion of n_j.

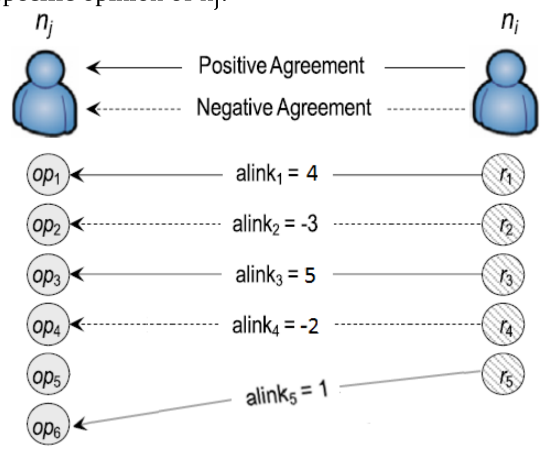

Fig. 1. example of relation of 2 users

They then calculated the users' knowledge level using three types of scores, the number of positive and negative scores, average of positive and negative scores and sum of positive and negative scores.

An explicit semantic analysis was used in study [2]. This work uses the selected dataset (Sina Microlab) as system input and converts it to a vector using TF-IDF[1] algorithm. Afterwards they used a proximity cosine to estimate similarity between users and then used an explicit semantic analysis to find experts from a particular definition in Bayesian networks. The general idea is to estimate an individual's x_i expertise in topic Q which results in $P(x_i|Q)$. To find this estimation, conventional methods, like lingual models, use various Bayesian network definitions but this work assumed that the value of $P(x_i |Q)$ is equal to that individual's expertise similarities in topic Q. Then they obtained "Follow" interactions which in turn allowed them to calculate user interaction, degree of collaboration and users' influence on each other's expertise. In the end, they used each user's similarity value and the score obtained in user interaction analysis to calculate the individual's final expertise.

Information overload is one of the aspects of Expert finding. Many techniques for reducing information overload is surveyed which can lead to finding interested group of people and thus finding experts out of them by implementing social network analysis methods on those groups and mine the experts [3]. In [4], users' reputation is considered as a factor for finding experts in Q&A forums. Two techniques are presented in [4], the first technique is based on asker and answerer's reputation in various threaded discussions and the second technique is based on user's answer quality which is based on category which the user participates in. In the first technique, co-existing users in discussion is extracted and modeled and in second technique semantic similarity among posts of co-existing users for a given topic is a basis for quality of answers. Three parameters are given for finding experts in a specific topic or answer in Yahoo Answer. These parameters are: user's knowledge profile, user reputation and link analysis [5].

This study, uses data mining to obtain the data related to each user's Q&A History and uses them to create a knowledge profile for each user. This profile, shows the topics of each user's knowledge. This information along with supplementary information, e.g. topic and date, are extracted from the dataset and then converted into vector space, in other words each user has a vector that corresponds to his/her knowledge. After creating user knowledge profile, each user is awarded a knowledge score, which depends on where that particular topic is usually mentioned (title, question description or answer) and how similar that particular topic is to the user's knowledge.

User Q&A record is used in various context for user reputation score derivations. This information includes the number of answers and the number of times that the answers were chosen as the best answer, as well. In link analysis, the users' network based on the links between users in the Q&A community is formed and then uses an algorithm to analyze user interaction. There are various algorithms that can be used for such purposes including the PageRank algorithm, HITS or one can use social network Metrics such as degree, closeness and betweenness centrality. Finally user expertise is calculated by summing these three scores.

[1] Term Frequency–Inverse Document Frequency

Work [6] showed that topic-based expert-finding can be achieved through Latent Topic Modeling. For example user expertise in a particular topic can be modeled based on the answers and comments that they provide on that particular topic. This method extracts user interests from user profiles and then finds the experts by using the questions and answers, obtaining the user score in each topic and using link analysis. The size of the dataset is particularly important in this method, since one cannot employ unsupervised or weakly supervised methods for user interest extraction when dealing with large datasets.

Study [7] proposed the CRAR[1] method, which obtains user authority using link analysis and does so based on the class or classes associated with that particular question and relevant topics (obtained from content analysis). In other words, first, user similarity is determined based on users' knowledge sharing logs, this user similarity is the basis of network creation. Next, link analysis is used on the network to rank expert users. This algorithm employs both content and ink analysis to find the experts.

Unlike previous studies, [8] uses methods other than the typical content and link-based analysis. Here they first introduce the conventional methods of expert-finding and then propose solving this problem by predicting the empty values in the adjacency matrix. The process of filling the matrix is employed in various fields, e.g. computer vision [9] and collaborative filtering using a weighted trace-norm regularizer [10]. In most cases the process is viewed from the perspective of filling matrix's empty values. The empty values can be filled in by using the adjacency matrix itself but this study used matrix rank function to show that this is an NP-hard problem (the rank of a matrix is the sum of its non-zero rows). Therefore, this problem becomes a nuclear norm optimization problem. They then used this norm to obtain the matrix values and user expertise.

Another problem in expert-finding is to find the exact number of people that have high expertise levels. For example assume that we are going to recommend K people as experts, then how, and on what basis, should we determine this K value. Moreover, the expert-finding problem can be considered by determining scores for each answer [11]. In [11], an automated method was designed by using the text and topic information within each answer. This information can be used to provide a classified model of answers and to determine the scores of answers at each level. The text information includes personal information, forum-relating information, the variety of characters and specific words in the text, and the statistical information pertaining to each answer.

The problem of methods which relies on users' Q&A history, is the sparsity of Q&A forum data. The problem of expert finding can be seen from the view point of learning ranking metric embedding. To find experts from this point of view, a novel ranking metric network learning framework was designed and then a random walk based learning method with recurrent neural network was developed to rank metric network embedding [12].

In [13], the focus was on evaluating a method which used *Learning to rank* (LTR) to rank feature vectors based on their relevance. This evaluation more concentrates on the quality of ranking function. The DRM method has been proposed in another resource [14]. DRM is a content analysis method which refers to the way that questions are suggested to the users and can be used to find the suitable users for answering any specific question. DRM is a probabilistic topic-sensitive method which uses the PLSA model to analyze each question's subject and then models each user in the roles of the asker and answerer based on his/her questions and answers. Implementation of this method is divided into two parts, the independent part and the dependent part. In the independent method (IDRM), user's cooperation in the topic are the only things that are taken into account but the dependent method (DDRM) also includes the relation and interactions between the users.

Using User interest to find experts is the focus of [15], which determines user interest from users' answer logs. They proposed two methods for expert-finding word-based and topic-based. The first method uses a lingual model and TF-IDF to model the users which assigns each question to one and only one topic category while the second method (also called STM[2]) can assign each question to multiple categories, which leads to more realistic models and better results.

In this paper we attempt to compare four expert-finding algorithms by applying them to an identical dataset. To this end we will introduce the dataset in the next section and then we will describe the conventional algorithms.

3. Dataset

We require a dataset relevant to Q&A communities, therefore we used the Stack Overflow dataset, which is a Q&A community in the Stack Exchange network. Stack Exchange is a network of 159 Q&A communities, each community in this network is created by experts and enthusiasts in a particular topic. Each of these communities consists of high quality questions and answers related to a particular topic and experts can be found in these communities based on user activities.

In the current paper the basis of knowledge in expert-finding is the ability to answer questions, Therefore the answers must be only about that particular question and should not include other discussions and personal opinions about that question. Stack overflow's questions and answers possess this key feature. We need to extract this kind of answers since unrelated discussions may take place in non-answer replies which, obviously, cannot be a reliable basis for evaluation of the users' knowledge.

This website revolves around the questions, when a user asks a question, the website creates a discussion around that question and sets the discussion to "open",

[1] Category Relevancy based Authority Ranking

[2] Segmented Topic Model

and other users can then answer that question. The question remains open until such time that the asker chooses one as the "best answer" (or until the best answer is chosen via the voting system), the discussion will be "closed" the moment that one of the conditions is satisfied and no one can post an answer afterwards.

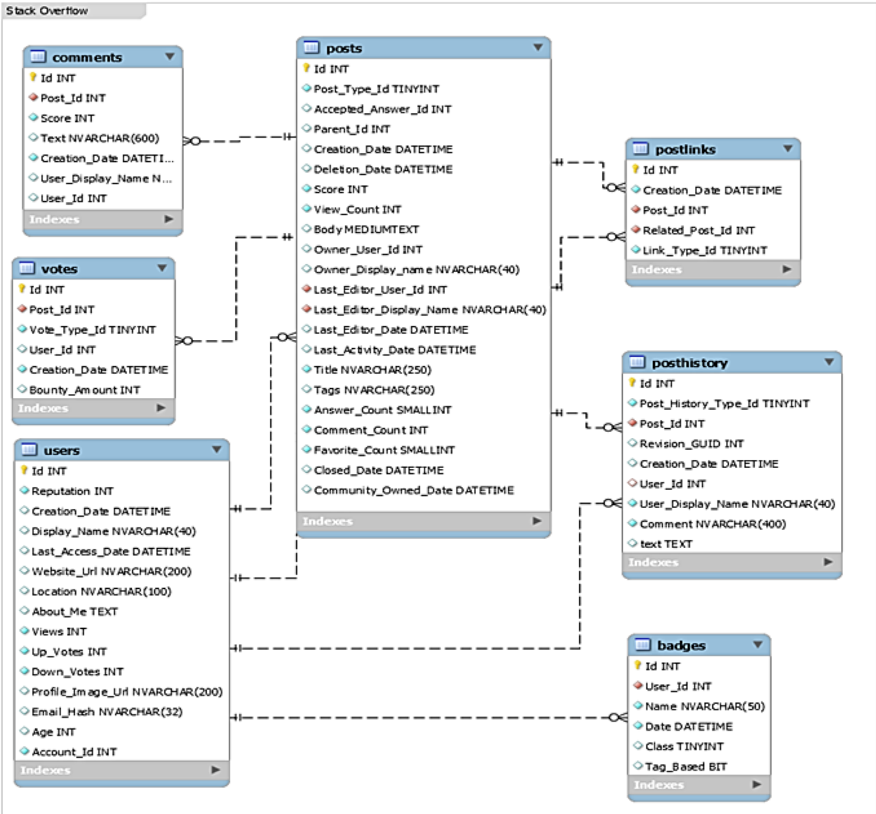

Fig. 2. Stack Overflow database scheme

We used the website's data extraction user interface in order to extract the required data. The information regarding the users, questions and answers, etc. were used to create database tables. Brief database scheme including important tables is illustrated in figure. 2.

As previously explained, to determine an individual's expertise, we need a clear and specific topic. In order to find the topics we used the "Tag" feature and found that C# and JavaScript were among the most discussed topics in Stack Overflow, which is confirmed by the community as well. The top 20 tags that generated the highest traffic are presented in figure. 3.

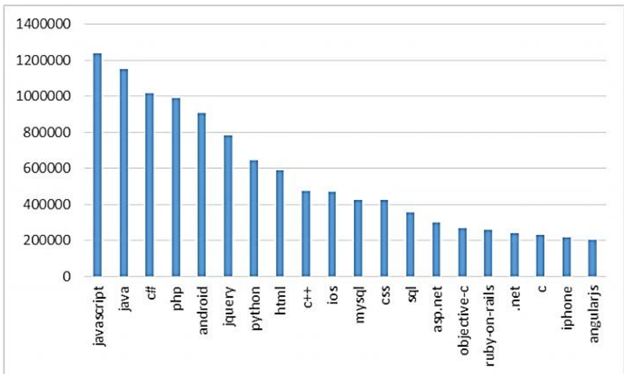

Fig. 3. Question count related to each tag in stack overflow

Therefore, we extracted the data related to the questions and answers about C# and JavaScript topics separately from 1st January 2014 to 1st January 2015. Following extraction and some preprocessing (including removing users that didn't have an ID and repetitive results due to similar tags), in the C# topic, we were left with 234877 answers from 89519 users while there were 354509 answers from 159408 users in JavaScript. After analyzing the dataset and extracting the desired data, we created table. 1 from the 2014 data regarding both JavaScript (JS) and C# tags.

Table 1. Data about extracted data

	No. of users	No. of questions	No. of answers	No. of questions without answer	Rate of answer
JS	159408	239316	354509	7.8%	86% By 25% of Users
C#	89519	167455	234877	9.1%	84% By 25% of Users

We also extracted score of each answer and used them in order to calculate score of each user in 2014 in each tag. These Scores will be used to evaluate algorithms' performance. Table. 2 shows a row of extracted data that illustrates an answer information in stack overflow.

Table 2. sample of extracted data

Questioner ID	Answerer ID	Question ID	Answer ID	Accepted Answer ID	Question Creation Date	Answer Creation Date	Answer Count	Question Score	Answer Score
2711395	474569	24362425	24362729	24363725	6/23/2014 9:21	6/23/2014 9:38	4	10	12
2711395	248703	24362425	24362812	24363725	6/23/2014 9:21	6/23/2014 9:42	4	10	3
2711395	343266	24362425	24363725	24363725	6/23/2014 9:21	6/23/2014 10:32	4	10	20
2711395	921321	24362425	24366141	24363725	6/23/2014 9:21	6/23/2014 12:37	4	10	2

4. Expert Finding Algorithms

As previously discussed there are various algorithms for expert-finding in social networks and the most notable ones are the PageRank, HITS, In-degree and Z-score algorithms. Next we will explain these algorithms and investigate their scoring methods.

4.1 Pagerank

This algorithm was first proposed in [16]. The goal of this algorithm was to determine the importance of various websites based on their referrals, but later it was used in various another context, including expert-finding, as well.

The PageRank algorithm is based on a user's random browsing between internet pages. The algorithm's idea is that a user starts in a website and randomly clicks on one of the links on that website and continues doing so during his/her internet session. Then the algorithm assigns importance to each page based on the number of views, but there is always the possibility that the user gets trapped in a loop of pages. To solve this problem, a jump possibility is added to the algorithm so that the user can jump out of the loop, this jump possibility is not limited to loop trap situations though and it has a fixed rate of happening at any given time though the jump possibility increases to 100% when the user is trapped in a loop. In other words the jump feature works like this:

- When the node is not connected to any other nodes, then there will be a jump.
- If there are other nodes connected to the present node, then there is a possibility 1-d (0<d<1) that the searcher jumps and consequently the possibility that it continues to the connected node is d. Here d is a damping factor and it is usually assumed to be equal to number between 0.85 to 0.9 in scientific studies.

Each we assume each node in the social network is a user, then each user's PageRank is calculated using Eq. (1), where PR(u) is Pagerank score of user u, d is the damping factor and represents the possibility of moving to the next connected node and therefore 1-d is the jump possibility, B_u is the set of nodes that are connected to node u, L(v) is the number of out-going edges of node v and N is total number of users.

$$\mathrm{PR(u)} = \frac{1-\mathrm{d}}{\mathrm{N}} + \mathrm{d}\Big(\sum_{\mathrm{v}\in \mathrm{B_u}} \frac{\mathrm{PR(v)}}{\mathrm{L(v)}}\Big) \tag{1}$$

4.2 HITS

This link analysis based algorithm has two parameters, hub and authority, that need to be updated every time that a node (user) or link (answer) is added. This algorithm also uses user interaction matrix and updates the matrix, hub and authority after an infinite repeat of the algorithm. To prevent diverging matrix values we need to normalize the values, and the algorithm provides this mechanism as well.

The HITS algorithm for expert-finding in Q&A communities assigns a hub and an authority parameter to each node and increases that node's "hub score" every time it asks a question while the "authority score" increase every time the node provides and answer. A node's authority is calculated by summing the hubs of every node that points to p (Eq. (2)) and hub of user p is equal to sum of the authority of users (i) that p points to them (Eq. (3)).

$$\mathrm{auth(p)} = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{hub(i)} \tag{2}$$

$$\mathrm{hub(p)} = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{auth(i)} \tag{3}$$

At first, when the user interaction network is formed, hub and authority values of every node is set to one and then Eq. (2) and (3) are used in order to update these parameters. A user's answers are essential to determining his/her expertise therefore authority will be used for final user ranking.

4.3 In-Degree

Degree is one of the centrality metrics in social networks and is equal to the edges entering the node. First we have to obtain the adjacency matrix of the Q&A network and the sum of the indices in a column is equal to the degree index, expect that here we set number of answers given by destination node (user) to starting node as edge weights. When using weighted edges, the degree metric is called "weighted In-degree" instead.

4.4 Z-Score

As it is obvious that answering multiple questions is a sign of expertise in a topic, asking questions is a sign that the user lacks knowledge. Therefore the Z-score method combines the Q&A Pattern of users. Here, assuming that the user has asked *q* questions and has provided a answers, a user's Z-Score is calculated by Eq. (4).

$$\mathrm{Z} = \frac{\mathrm{a}-\mathrm{q}}{\sqrt{\mathrm{a}+\mathrm{q}}} \tag{4}$$

Obviously the Z-score would be zero if the number of questions and answers are equal, it would be positive if *a* is greater than q and would be negative in other cases. This value is called the "Z number" when one uses the number of questions and answers while it is called the "Z degree" when one uses the In-Degree of the node as *a* and Out-Degree as *b*.

4.5 Algorithm Analysis

The PageRank algorithm is very sensitive to the knowledge environment. For example if individual B answers to individual A's question and individual C answers a question posed by B, then C has to score higher in expertise since he/she could answer the relative expert from the former interaction. But the question posed by A may have been in the C# field while the question posed by B may have been in JavaScript, which negates the previous conclusion. Therefore the analysis must be confined to a single topic which is exactly what we have done in this paper for two different datasets.

As it is mentioned in [11], there are situations where the HITS algorithm gives erroneous results as well. Take figure. 4 as an example, nodes 1, 5 and 10 should receive the highest scores but HITS algorithm gives high authority scores to 1, 6, 10, 11 and 12 and gives a score close to zero to node 5. This occurs since node 13 is connected to node 10 (high authority node) therefore, node 13 has a high hub score, which in turn increases the authority scores of nodes 6, 11 and 12. on the other hand, nodes which are connected to node 5 have low hub scores (relative to nodes connected to 1 and 10), therefore node 5 gets a low authority score.

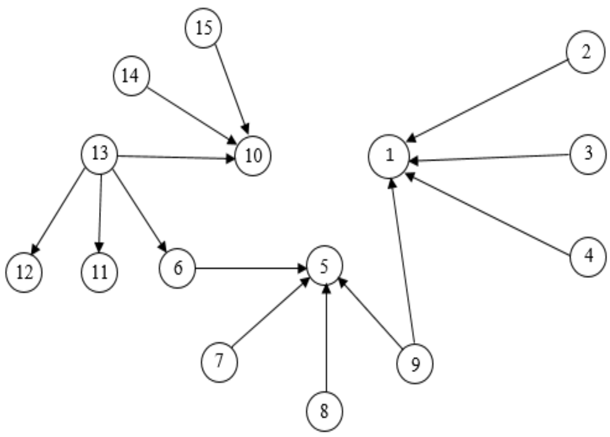

Fig. 4. Example of HITS algorithm

Then we should apply the algorithms to the information extracted from the dataset (i.e. the questions and answers) and obtain each user scores based on each algorithm. The explanation for the implications of each algorithm is as follows.

The PageRank algorithm inherently assigns higher importance to nodes that have higher number of reference (the PageRank algorithm does not take into account the quality of a reference because the definition of a reference quality refers to context of the reference which is not SNA approach) and in the context of Q&A communities, this means that the user has been active in answering the questions which are the basis of user expertise. Authority in HITS depends on the number of connections to other nodes and, here, it is a representation of the answers given by the node and even questions which this node has answered. As explained previously In-

The Z-score algorithm assigns user expertise score by using the concept of questioning and answering rate prediction and calculating users' standard deviation from that prediction. This is a purely mathematical method and its analysis is not based on social networks' structures and concepts, which is not favorable, but we can get good results by combining the Z-score algorithm with Degree Centrality metric.

In-degree is the simple metric in social network analysis, yet it also gives us a good understanding of the social network. In this context In-degree represents the number of unique users that the node (user) has given answer to. To improve this approach, we created weighted edges by adding weights to each edge that connect any two nodes. A weighted In-degree represents the number of answers a user has given to others in that particular dataset and time period.

5. Implementation

First, we have to define the user interaction network. A network consists of nodes and edges and here the nodes represent the users while the edges represent the answers given by the users. Figure. 5 shows a part of the user interaction network and the top 10 users as reported by the In-degree algorithm. Node A has the highest In-degree in the user interaction network, hence it is also the biggest and darkest node in the network. It should be noted that this network is an asker-answerer network and edges are drawn from asker to answerer.

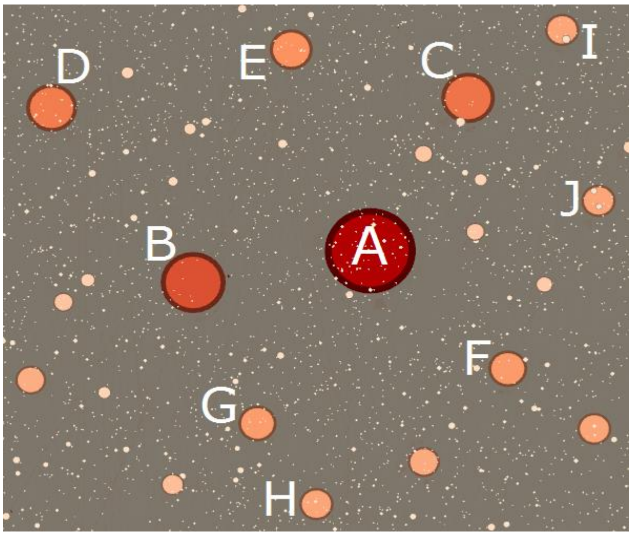

Fig. 5. Part of User Interaction Network

Table 3. Scores obtained by top 10 users in C#

Symbol	UserID	PageRank	In-Degree	Weighted In-Degree	HITS (Authority)	Z Number	Z Degree
A	3010968	0.0025	1845	2102	0.00676	43.5	40.5
B	22656	0.0036	1357	1452	0.00497	37.9	36.6
C	284240	0.0014	1090	1191	0.00399	33.0	31.6
D	1159478	0.0160	1063	1143	0.00390	33.7	32.5
E	993547	0.0019	894	939	0.00328	30.4	29.7
F	301857	0.0059	807	871	0.00296	29.5	28.4
G	470005	0.0010	794	850	0.00291	29.1	28.1
H	1081897	0.0011	709	743	0.00260	27.2	26.6
I	1197518	0.0009	708	756	0.00259	27.4	26.6
J	23354	0.0018	707	774	0.00259	27.4	26.3

Degree is a measure of the number of entering edges and in this context it represents the number of unique individuals the user has answered. Z-number depends on the number of answers and questions extracted from the dataset. the algorithm assigns negative values to the Z-number when question count passes answer count of each user while providing more answers by the user increases the score. Z-Degree, assigns positive values to In-degree and negative values to Out-degree and determines expertise based on those values.

Table. 3 shows the scores obtained by 10 different users. The marked scores are the highest scores in each algorithm (meaning these scores are obtained by the most expert users as indicated by that particular algorithm). For example, user 1159478 has a score of 0.0160 in PageRank algorithm while the most expert user as predicted by In-degree, weighted In-degree, HITS, Z-Number and Z-degree algorithms was user 3010968 which obtained scores of 1845, 2102, 0.00676, 43.5, 40.5 respectively.

6. Evaluation

We removed users that had less than 50 answers to analyze the more active users to do the evaluation process. In order to evaluate each algorithm, we sorted the top 50 users by score they achieved based on each algorithm in mentioned time period in each Tag. We also sorted the top 50 users, based on scores they achieved according to their answers in the Stack Overflow in mentioned time period in each Tag. Then we compared the algorithm predicted top 50 with the top 50 that was extracted from Stack Overflow and determined how many common users are in each 50. The reason that we chose top 50 users is the fact that in specific tags in stack overflow, score of people suddenly decrease at some points and we can say for sure that top 50 users in each tag answer a large amount of questions.

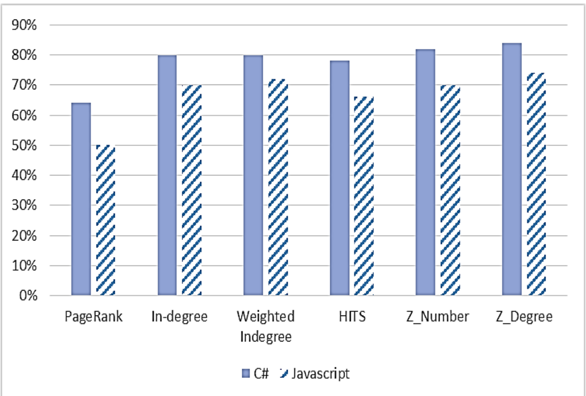

Fig. 6. Performance of each method

Figure. 6 shows the ratios of users that each algorithm successfully predicted, obviously, a higher ratio means that particular algorithm has had a better performance. In other words this figure shows the accuracy (in percentage) of each algorithm in finding experts in C# and JavaScript areas in Stack Overflow in 2014.

As can be seen in figure. 6, Z-degree algorithm had a ratio of 84% which translates to the best performance among all algorithms. All algorithms except the PageRank algorithm gave almost similar results since they solely focus on the answers provided by the user. The score given to each user in the dataset is based on the users' opinions and does not take into account that "a user should get more credit when he/she answers the questions of people who have answered to another expert's questions previously". Stack Overflow dataset scores rank the people based on scores they receive. The scores come from up-votes they take by other users and that up-vote is naturally come from quality of answer, reputation and being best answer by questioner's choice. That is why PageRank has had the worst performance. The reason that the Z-score algorithm had the best performance between the other 3 algorithms is that this algorithm also takes into account the questions posed by the user and treats them as an indication that he/she lacks knowledge.

7. Conclusion

This paper compared various expert-finding algorithms in an online Q&A Forum. One Million Questions and Answers were extracted and pre-processed in order to use in expert-finding algorithms. It was shown that performance of each algorithm depends on scenario and structure of network. Each of these algorithms assigns scores to users in their own unique way and then reports the user with the highest score as the expert. To achieve our goal, first, we discussed the various algorithms and then extracted the desired datasets from the Stack Overflow database and prepared the data for pre-processing. Then algorithms were applied to the users' questions and answers and results were compared with the scores given in the community itself in order to obtain each algorithm's performance. The results showed that the Z-Degree algorithm had the highest percentage of expert-finding therefore it also had the best performance. Overall Z-Number, In-Degree, HITS and Pagerank are in the next ranks, respectively.

References

[1] Wei, Chih-Ping, Wen-Ben Lin, Hung-Chen Chen, Wen-Yu An, and Wei-Chang Yeh. "Finding experts in online forums for enhancing knowledge sharing and accessibility." Computers in Human Behavior 51 (2015): 325-335.

[2] Li, Xiu, Jianguo Ma, Yujiu Yang, and Dongzhi Wang. "A Service Mode of Expert Finding in Social Network." In 2013 International Conference on Service Sciences (ICSS), pp. 220-223. IEEE, (2013).

[3] Mittal, Prathma, Shivangi Garg, and Sunita Yadav. "Social Network Analysis using interest mining: A critical review." Indian Journal of Science and Technology 9.16 (2016).

[4] Faisal, Muhammad, Ali Daud, and Abubakr Akram. "Expert Ranking using Reputation and Answer Quality of Co-existing Users." International Arab Journal of Information Technology (IAJIT) 14.1 (2017).

[5] Liu, Duen-Ren, Yu-Hsuan Chen, Wei-Chen Kao, and Hsiu-Wen Wang. "Integrating expert profile, reputation and link analysis for expert finding in question-answering websites." Information Processing & Management 49, no. 1 , 312-329. (2013).

[6] Zhou, Guangyou, Siwei Lai, Kang Liu, and Jun Zhao. "Topic-sensitive probabilistic model for expert finding in question answer communities." InProceedings of the 21st ACM international conference on Information and knowledge management, pp. 1662-1666. ACM, (2012).

[7] Li, Xiu, Jianguo Ma, Yujiu Yang, and Dongzhi Wang. "A Service Mode of Expert Finding in Social Network." In 2013 International Conference on Service Sciences (ICSS), pp. 220-223. IEEE, (2013).

[8] Zhao, Zhou, Lijun Zhang, Xiaofei He, and Wilfred Ng. "Expert finding for question answering via graph regularized matrix completion." IEEE Transactions on Knowledge and Data Engineering 27, no. 4, 993-1004. (2015).

[9] Peng, Yigang, Arvind Ganesh, John Wright, Wenli Xu, and Yi Ma. "RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images." IEEE Transactions on Pattern Analysis and Machine Intelligence 34, no. 11 (2012): 2233-2246.

[10] Srebro, Nathan, and Ruslan R. Salakhutdinov. "Collaborative filtering in a non-uniform world: Learning with the weighted trace norm." In Advances in Neural Information Processing Systems, pp. 2056-2064. 2010.

[11] Le, Long T., et al. "Evaluating the Quality of Educational Answers in Community Question-Answering." Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries. ACM, (2016).

[12] C Zhao, Zhou, et al. "Expert finding for community-based question answering via ranking metric network learning." Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence. AAAI Press, (2016).

[13] Beierle, Felix, et al. "Finding the Right Experts and Learning to Rank Them by Relevance: Evaluation of a Semi-Automatically Generated Ranking Function." LWA. (2014).

[14] Xu, Fei, Zongcheng Ji, and Bin Wang. "Dual role model for question recommendation in community question answering." In Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval, pp. 771-780. ACM, (2012).

[15] Riahi, Fatemeh, Zainab Zolaktaf, Mahdi Shafiei, and Evangelos Milios. "Finding expert users in community question answering." In Proceedings of the 21st International Conference on World Wide Web, pp. 791-798. ACM, (2012).

[16] Page, Lawrence, Sergey Brin, Rajeev Motwani, and Terry Winograd. "The PageRank citation ranking: bringing order to the web." (1999).

Ahmad A. Kardan received his BS in Electrical Engineering from Sharif University of Technology (1976-Iran), his M.Sc. in Digital Systems from the Brunel University (1997UK), and his PhD in Bio-Electric Engineering from Imperial College of Science and Technology (2001-UK). He founded E-Learning Center of Amirkabir University of Technology in2000and managed the center till 2004.He is currently a faculty member of the Computer Engineering Department, at Amirkabir University of Technology, Tehran, Iran. He teaches graduate courses in computing and information technology with emphasis on advanced e-learning and distributed educational systems. Dr Kardan is involved in researches in Intelligent Tutoring Systems (ITS), Learning Advisory Systems, Adaptive Learning, Learner Modeling, Concept Mapping, Collaborative Learning, Annotation Processing and Applying Data Discovery in e-Learning. He has presented more than 80 papers at national and international conferences, journals and as chapters for related books.

Behnam Bozorgi received his BS in Software Engineering from ferdowsi university of mashhad (2014-Iran), his M.Sc. in Information Technology from Amirkabir University of Technology (2016-Iran). His area research interests include user modeling, social network analysis, expert-finding.