

Article

# Exploring Trusted Relations among Virtual Interactions in Social Networks for Detecting Influence Diffusion

Heba M. Wagih <sup>1,\*</sup>, Hoda M. O. Mokhtar <sup>2</sup> and Samy S. Ghoniemy <sup>3</sup>

<sup>1</sup> Information Systems Department, The British University in Egypt, El Shorouk City, Cairo 11837, Egypt

<sup>2</sup> Information Systems Department, Cairo University, Cairo 12613, Egypt; h.mokhtar@fci-cu.edu.eg

<sup>3</sup> Computer Networks Department, The British University in Egypt, El Shorouk City, Cairo 11837, Egypt; samy.ghoniemy@bue.edu.eg

\* Correspondence: heba.wageeh@bue.edu.eg

Received: 12 July 2019; Accepted: 9 September 2019; Published: 16 September 2019

**Abstract:** Recently, social networks have shown huge potential in terms of collaborative web services and the study of peer influence as a result of the massive amount of data, datasets, and interrelations generated. These interrelations cannot guarantee the success of online social networks without ensuring the existence of trust between nodes. Detecting influential nodes improves collaborative filtering (CF) recommendations in which nodes with the highest influential capability are most likely to be the source of recommendations. Although CF-based recommendation systems are the most widely used approach for implementing recommender systems, this approach ignores the mutual trust between users. In this paper, a trust-based algorithm (TBA) is introduced to detect influential spreaders in social networks efficiently. In particular, the proposed TBA estimates the influence that each node has on the other connected nodes as well as on the whole network. Next, a Friend-of-Friend recommendation (FoF-SocialI) algorithm is addressed to detect the influence of social ties in the recommendation process. Finally, experimental results, performed on three large scale location-based social networks, namely, Brightkite, Gowalla, and Weeplaces, to test the efficiency of the proposed algorithm, are presented. The conducted experiments show a remarkable enhancement in predicting and recommending locations in various social networks.

**Keywords:** social networks; influential nodes; recommendation systems

## 1. Introduction

A noticeable focus in research activities is devoted to answering the question of how to efficiently detect the most influential nodes in social networks. Influential nodes play a crucial role in understanding the behavior of information diffusion. The success of many different applications, such as viral marketing, recommendation applications, and decision-making applications, relies on discovering and using influential nodes. Discovering influential nodes is a challenging research area in social networks. Many research works have been proposed to detect the influence of each node in a network, such as the Degree Centrality Measure (DCM) [1], Betweenness Centrality Measure (BCM) [1], Closeness Centrality Measure (CCM) [1], and K-Core decomposition [2], amongst others.

DCM is a type of centrality measure that retrieves the number of neighbor nodes at a distance of exactly one hop. The higher the degree of the node, the more central the node is considered. Social networks consider people with a higher degree to have more influence on the network. BCM is a type of centrality measure that determines the degree by which nodes act as a bridge between two other nodes. CCM is another type of centrality measure that determines how long it takes to diffuse information from a node to other nodes in the network, and this measure has a noticeable impact in

viral marketing [1]. In K-Core decomposition, a graph  $G$  is divided into a number of subgraphs known as K-Core subgraphs where all nodes in each K-Core subgraph have a degree of at least  $k$ . Inspired by the importance of detecting influential nodes, several works have been conducted to this end.

Different approaches for investigating the influence dispersion in social networks have been discussed [3–5]. Bonchi presented a vital review that summarizes several algorithms, which were introduced for identifying influential spreaders, and defined the information propagation mechanisms in social networks [6]. Zhao et al. proposed the Local Centrality with a Coefficient (CLC) algorithm to detect influential nodes [7]. CLC considers both the topological connections between nodes as well as the number of each node's neighbors. A decreasing function called the coefficient of local centrality was constructed to consider relations with a depth of four levels of neighbors.

Detecting influential nodes has a clear role in recommendation systems. Thus, several recent works have been introduced to study the effect of detecting influential nodes in the recommendation process, as presented in [8–10]. Peng et al. presented an evaluation model to measure both direct and indirect propagation of social influence in social networks [11]. Other approaches for improving recommendation services focus on studying the behavioral trust between nodes and its impact on the information diffusion power, as in [12,13]. Adali et al. proposed a quantifiable measure of trust, which represents the communication behavior between social network participants [12]. The authors categorized behavioral trust in terms of conversational trust and propagation trust. Additionally, in [13], Song et al. introduced a novel recommendation model based on trust relations and item ratings named TrustTR. The authors built their model based on trusted friends' recommendations, item reputation, and user history ratings, and experimentally proved that their model shows better accuracy compared to other prior recommendation models.

Location recommendation systems (LRSs) are a new type of recommender system that have greatly benefitted from the detection of influential nodes. Bao et al. introduced a location-based and preference-aware recommendation system that suggests nearby locations [14]. The proposed system is modeled using users' preferences derived from their check-in history and social opinions from local domain experts who share common interests. The authors did not address the friendship connections in their research. This drawback was addressed in work introduced by Hakan et al. [15] and Hangal et al. [16]. Hakan et al. proposed a random walk-based context-aware friend recommendation algorithm (RWCFR) [15]. The proposed algorithm considered the current context (i.e., social ties and current location) for the users to provide recommendations. The authors employed a random walk with a restart approach to rank the recommendation scores of the users for recommendation. Although the authors considered the social interactions among users, their algorithm lacks the trust calculation among friends. Hangal et al. proposed an algorithm for enhancing recommendations by considering both weighted and directed edges in the social graph [16]. They proved experimentally that their proposed algorithm showed an improvement in recommendation accuracy by approximately 35%.

Despite many research works having been conducted to consider social connections and visited locations when providing a location recommendation, none of these have focused on studying the strength of social connections among social network participants and the possibility of recommendations from friends of friends rather than direct friends. In the proposed algorithms, a trust-based algorithm (TBA) and a Friend-of-Friend recommendation (FoF-SocialI), we address these gaps by first analyzing the social connections in terms of trust and influence between connected friends, which provide deeper insights into location recommendation services. Secondly, we consider up to five levels of friendship relation between the network participants. The proposed algorithms have remarkably enhanced location recommendation services. The experiments show an improvement in the recommendation accuracy by approximately 75%.

Motivated by the significant influence of these research directions in many application areas, we followed efforts introduced in previous works and presented the following contributions:

1. We investigated the role of influential nodes in the recommendation process and more specifically in location recommendation.

2. We examined the effectiveness of the algorithm proposed earlier in [17] in detecting the influential nodes compared to other conventional algorithms.
3. We enhanced the algorithm [17] by introducing the importance function (Imp) and the Reverse set of Influential Node function (RIN).
4. We extended the algorithm [17] by proposing a novel Friend-of-Friend (FoF) location recommendation algorithm that computes location recommendations employing FoF relationship information.

The focus of this investigation was to study and compute trust in social networks rather than a study of the network literature. The remainder of this paper is organized as follows: Section 2 highlights works related to influential node detection and its influence in various application domains. In Section 3, we revise the published algorithm [17] and introduce a modified version of the recommendation algorithm that considers Friend-of-Friend relations. In Section 4, we compare and evaluate the proposed algorithms with other conventional algorithms. Finally, in Section 5, we conclude this paper and propose directions for future work.

## 2. Related Work

The pervasiveness of social network applications has a great amount of influence over people in the modern world. Today, people can share their ideas, thoughts, and knowledge with their friends, relatives, and acquaintances easily through social networks. Recently, a large amount of machine data have appeared due to the use of social network applications in different domains, such as recommendation systems, viral marketing, epidemic control, and others. Those recent applications have attracted the attention of both computer scientists and physical societies. Consequently, new research challenges have arisen, addressing such problems as influence maximization, community detection, and extraction of vital nodes.

In general, social networks are an abstract representation of social systems, where users interact and share ideas and information. Thus, it is crucial to identify influential nodes in social networks. Detecting influential nodes is a challenging research area, and many influential detection algorithms have been proposed to disclose the centrality weight of each node in the network.

Recent investigations have introduced different techniques for detecting influential nodes [18,19]. Narayanam et al. introduced both the Maximum Degree Heuristic and the High Clustering Coefficient Heuristic [20]. The Maximum Degree Heuristic computes the degree of each node in the network and chooses the top-K nodes with the highest degree as the key influencers. The High Clustering Coefficient Heuristic computes the clustering coefficients of the nodes in the network and chooses the top-K nodes with the highest clustering coefficient as the key influencers.

Rossi et al. introduced an enhanced version of the K-Core decomposition algorithm, known as the K-truss subgraph [21]. The K-truss is a more decomposed graph than that developed by employing the K-Core algorithm. The K-truss reduces the number of information diffusers in the network, which consequently influences a more significant part of the network. Of particular importance is the work proposed in [22], which stressed that the topological properties of the nodes play an essential role in understanding their spreading capabilities. The authors introduced a triangle-based extension of the core decomposition of graphs for identifying influential nodes based on the properties of K-truss decomposition. Using six different online social networks, the authors experimentally proved that nodes belonging to the maximal K-truss subgraph have better diffusion behavior. Basuchowdhuri et al. introduced an algorithm for detecting influential nodes using a minimum K-Hop dominating set [23]. Basuchowdhuri et al. built their algorithm on the idea of deciding a maximum number of k-Hops to diffuse information among all the nodes of the graph. They compared their algorithm with other betweenness centrality measures, and they experimentally proved that influential nodes have higher betweenness values.

An improved version of degree centrality, named clustered local-degree (CLD), was introduced in [24]. The CLD of a given node is the product of the sum of the nearest neighbors' degrees and its clustering coefficient. The authors performed their simulation on eight real networks to compare the

performances between the CLD centrality and another six measures. They experimentally showed that CLD has a competitive performance in detecting influential nodes.

Detecting influential nodes has a crucial role in improving many applications. Goyal et al. proposed a framework for enhancing advertisement activities based on detecting the influential nodes [25]. The authors used discrete time models to compute the trust level between users upon which they can discover the information spreaders. Most recently, Davoudi et al. introduced an algorithm for modeling an accurate recommender system [26]. The proposed algorithm aims to find the relevant items for each user based on his/her interests as well as the social interactions between users. The authors used the probabilistic matrix factorization method to predict user rating for products and rating-based and connection-based similarity measurements for modeling similarity.

Community detection is another challenge that was found to be very tempting in social networks. Community detection is a complex task that benefits from detecting influential nodes; therefore, many community detection algorithms have been introduced, such as in [27,28]. Yang et al. proposed a model called BIGCLAM for detecting overlapping communities that consist of millions of nodes and edges [27]. The proposed algorithm can detect both overlapping and non-overlapping communities. Massive experiments were performed to evaluate the proposed algorithm using six different social networks. The authors experimentally proved that BIGCLAM outperforms the previous work regarding discovering network communities. Wei et al. proposed a method for identifying influential nodes in communities [28]. The proposed algorithm is based on the BIGCLAM model. The influential diffusion capacity is determined based on the number of communities to which the node belongs, and the propagation speed is determined by the network constraint coefficient of the node. The susceptible infected recovered (SIR) model [29] is used to evaluate the performance of the proposed method.

### 3. Location Recommendation Based on Social Trust

Recommendation Systems play an important role in providing better customized user experiences. Many research have been introduced to enhance the recommendation quality by addressing problems such as information overload and information heterogeneity in social networks. Different recommendation systems have been developed to provide users with personalized recommendations on items such as movies, books, news, and places to visit. Lately, a new type of recommendation system has been introduced; these are known as the Location Recommendation Systems (LRSs). The primary purpose of LRSs is to recommend new places to be visited based on the user's check-in history, points of interest, and social relations obtained from location-based social networks. The recommendation problem in location-based social networks has attracted remarkable attention in research and practical applications. The primary purpose of recommending new places is to provide a suggestion to a user to visit new places or points of interest. This type of recommendation is essential, notably for those who are traveling to a new city and want the most satisfying experience during their trip.

In this section, we revise the trust-based location recommendation algorithm (TBA) that was introduced in [17] and propose a new algorithm for enhancing location recommendation activities, namely friend-of-friend influence based location recommendation algorithm (FoF-SocialI) that is an extended version of TBA. The proposed algorithms provide location recommendation services by considering both users' locations as well as the friendship degree between users. Friendship relationships are the driving force in the recommendation process. People prefer to receive recommendations from those to whom they are strongly tied. We define the strength of social ties in terms of trust and influence. For each pair of connected friends, there is a certain degree of interpersonal trust, which is defined by the weight of the relation between the connected friends. In some cases, trust cannot by itself guide the recommendation process, where  $friend_i$  can trust  $friend_j$  with a high percentage in recommending a particular item and this percentage decreases noticeably if the item type or category changes. To solve this problem, we introduce a new metric named influence.

We model influence based on social interests, preferences, and choices. The higher the commonalities in interests and preferences, the more the influence a friend has on his/her connected friends.

### 3.1. TBA: Trust-Based Location Recommendation Algorithm

In this section, we discuss location-based recommendation algorithms that consider the trust between nodes in the recommendation process. In [17], we proposed a trust-based measuring algorithm (TBA) that offers location recommendations for direct friendship relations based on users' geospatial locations. The proposed algorithm is applied on directed graphs that represent friends in social networks.

In the following discussion, we highlight the main contributions presented in [17]. TBA consists of two main modules: the trust calculation module and the influence calculation module.

#### 3.1.1. Trust Calculation Module

Creating a web of trusted relationships is a new research topic with many challenging and interesting questions. In this work, we show that the mutual trust between network participants is the main factor that affects the influence diffusion in social networks.

Consider a social network with several interrelated connections between three friends. Figure 1 shows an example of weighted relation social network.

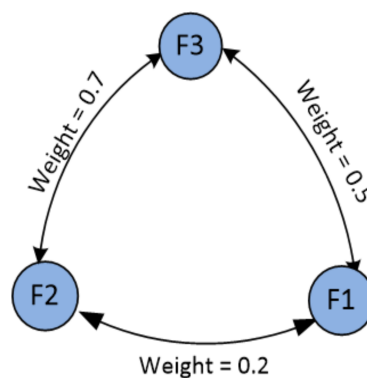


Figure 1. Example of weighted social network.

In Figure 1, for each pair of friends ( $f_i$  and  $f_j$ ), a *Weight* value that represents the correlation between the friends' attributes is computed using Pearson Correlation Coefficient (PCC) [30], and is given by Equation (1):

$$\text{Weight}(f_i, f_j) = \text{PCC}(f_{i\text{attr}}, f_{j\text{attr}}). \quad (1)$$

Each friend that participates in the social network has a specific trust degree value on his/her connected friends known as *EdgeTrust* and is given by Equation (2):

$$\text{EdgeTrust}(f_i, f_j) = \frac{\text{Weight}(f_i, f_j)}{\sum_{x=1}^m \text{Weight}(f_i, f_x)}, \quad x = 1, 2, 3, \dots, j, \dots, m. \quad (2)$$

where  $i$  is the current node,  $j$  is any other node in the network, and  $m$  is the index of the last node which consequently denotes the network size  $|m|$ . In Equation (2),  $\text{Weight}(f_i, f_j)$  denotes the weight value of the relation between  $\text{friend}_i$  and  $\text{friend}_j$  and  $\text{Weight}(f_i, f_x)$  denotes the weight value of every relation between  $\text{friend}_i$  and any directly related friends in the social network (i.e., there exists an edge between  $\text{friend}_i$  and each  $\text{friend}_x$ ).

On analyzing social networks, we discovered that friends with low trusted values might have a negative influence on their connected friends. Thus, we pruned all relationships with trust values less than a certain threshold by applying the *importance function* (*Imp*), building a trusted web network:

$$Imp(EdgeTrust(f_i, f_j)) = \begin{cases} 1 \geq Threshold, \\ 0 < Threshold. \end{cases} \quad (3)$$

Furthermore, for each friend in the network, we assume a node trust value that this friend receives from directly connected friends, which is known as *NodeTrust* and is given by Equation (4):

$$NodeTrust(f_i) = \sum_{x=1}^m EdgeTrust(f_x, f_i), \quad x = 1, 2, 3, \dots, j, \dots, m. \quad (4)$$

In Equation (4), the higher the node trust value, the more trustworthy this friend (*friend<sub>i</sub>*) is considered on the social network.

### 3.1.2. Influence Calculation Module

Despite the noticeable effect of trust in predicting the influence friends might have on each other, in some cases, the trust and confidence between a pair of friends are high, but the influence is low. This can happen for behavioral, geographical, and psychological reasons, all of which have a notable influence on users' interests. In the proposed work, we examine two types of influence: edge influence and node influence.

We assume that each friend in the social network has a particular influence on all directly related friends, which is known as the *EdgeInfluence* (*EdgeInf*) between each pair of friends. Herein, we consider the influence *friend<sub>i</sub>* has on *friend<sub>j</sub>* to be the product of the trust *friend<sub>j</sub>* has on *friend<sub>i</sub>* and the recommendation weight between *friend<sub>i</sub>* and *friend<sub>j</sub>* as shown in Equation (5):

$$EdgeInf(f_i, f_j) = EdgeTrust(f_j, f_i) * RecW(f_i, f_j). \quad (5)$$

The recommendation weight (*RecW*) is the extent to which a pair of friends might have mutual interests, and this can be predicted by analyzing the user's check-in data. Each check-in value is represented in terms of spatial coordinates: the longitude and latitude that define the visited place. *RecW* is calculated as the ratio between the visited places (*VP*) of *friend<sub>i</sub>* and *friend<sub>j</sub>* and the total number of visited places (*TVP*) of *friend<sub>i</sub>*, as shown in Equation (6):

$$RecW(f_i, f_j) = \frac{VP(f_i, f_j)}{TVP(f_i)}, \quad (6)$$

$$TVP(f_i) = \{|VP(f_i, X)|\}, \quad X \text{ is the set of all visited places.} \quad (7)$$

Following the social network given in Figure 1, Figure 2 shows an example of a recommendation weight calculation.

Despite the importance of edge trust in calculating the edge influence among the social network participants, in some cases, the recommendation weight has a more significant impact in determining friends with the highest edge influence, as shown in Table 1. Table 1 shows the edge trust and edge influence values between social network participants, as given in Figure 1.

In Table 1, although Friend 2 has a higher trust value in Friend 3 because of the importance of the spatial factor presented in the visited locations by the three Friends, Friend 1 shows more influence over Friend 2.

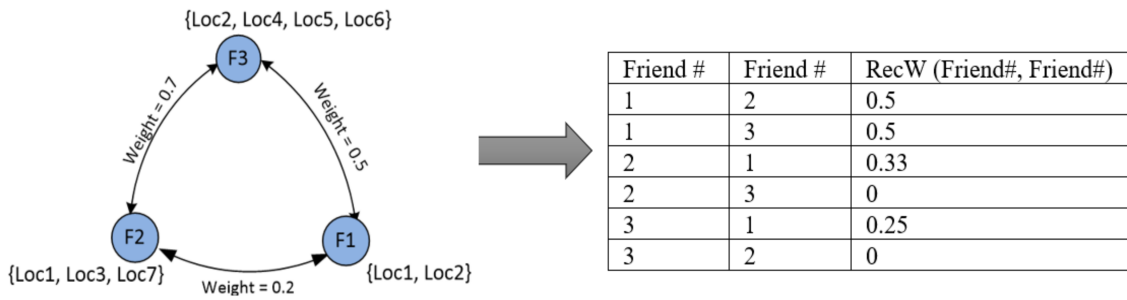


Figure 2. Example of a recommendation weight calculation.

Table 1. Edge trust and edge influence among different friends in social networks.

Friend#	Friend#	Edge Trust	Edge Influence
Friend 1	Friend 2	0.285	0.11
Friend 1	Friend 3	0.71	0.205
Friend 2	Friend 1	0.22	0.094
Friend 2	Friend 3	0.77	0
Friend 3	Friend 1	0.41	0.177
Friend 3	Friend 2	0.58	0

On the basis of the calculated edge influence, the friend with the highest edge influence is the recommended one for the location recommendation process, as shown in Equation (8):

$$LocRec(f_i) = f_j \mid f_j \in F \wedge EdgeInf(f_j, f_i) \geq EdgeInf(f_z, f_i), \forall f_z \in F \wedge z \neq j, \tag{8}$$

where ( $F$ ) is the set of all directly connected friends.

Each friend in the social network has an influence degree on the whole network known as *NodeInfluence* (*NodeInf*) and is defined by Equation (9):

$$NodeInf(f_i) = \sum_{x=1}^m EdgeInf(f_i, f_x) * NodeTrust(f_i). \tag{9}$$

Although the estimated weight for each pair of friends in the social network,  $friend_i$  and  $friend_j$ , is invariant because of the asymmetric property of friendship relations, some behavioral characteristics such as trust and influence vary in their value. For example, the trust value  $friend_i$  has on  $friend_j$  might differ from the trust  $friend_j$  has on  $friend_i$ ; this is similar for the influence case.

Attaching everything together, Equation (1) to Equation (9) represent the building blocks of the proposed algorithm as given in Algorithm 1.

Algorithm 1 solves three types of queries: Firstly, it retrieves the friends’ influence on other connected friends in a social network; secondly, it retrieves the friends’ influence on the whole social network, and, thirdly, it detects the reverse set of influential Nodes (RIN) of each node in the social network for location recommendation as given in Equation (10):

$$RIN(u) = v_i \mid v_i \in V \wedge EdgeInf(u, v_i) \geq EdgeInf(v_j, v_i), \forall v_i \in V \wedge \forall v_j \in V. \tag{10}$$

**Algorithm 1:** Trust-Based Location Recommendation Algorithm

---

```

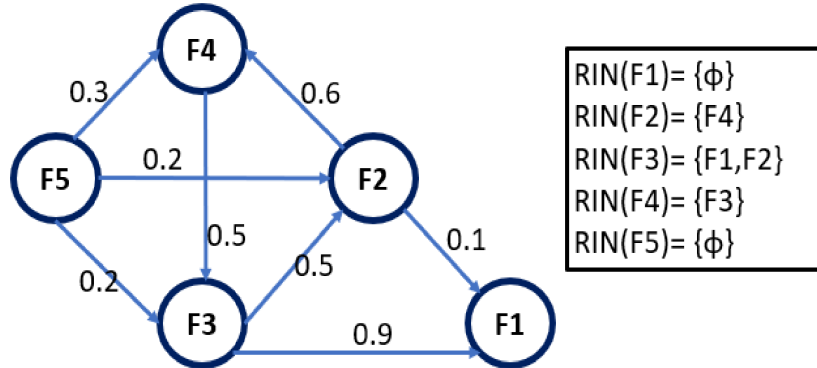
1: for each friend  $f_i$  in Social Network Do
2:   for each friend  $f_j$  in Social Network Do
3:      $Weight(f_i, f_j) = cor(f_{iattr}, f_{jattr})$ 
4:      $EdgeTrust(f_i, f_j) = \frac{Weight(f_i, f_j)}{\sum_{x=1}^m Weight(f_i, f_x)}$ 
5:      $NodeTrust(f_i) = \sum_{x=1}^m EdgeTrust(f_x, f_i)$ 
6:     for each visited place  $x$  in Relation  $VP(f_i, x)$  Do
7:        $TVP(f_i) = \{|VP(f_i, x)|\}$ 
8:        $RecW(f_i, f_j) = \frac{VP(f_i, f_j)}{TVP(f_i)}$ 
9:     end for
10:     $EdgeInf(f_i, f_j) = EdgeTrust(f_j, f_i) * RecW(f_i, f_j)$ 
11:     $LocRec(f_i) = \{f_j \in F \mid EdgeInf(f_j, f_i) \geq EdgeInf(f_z, f_i), \forall f_z \in F\}$ 
12:  end for
13:   $NodeInf(f_i) = \sum_{x=1}^m EdgeInf(f_i, f_x) * NodeTrust(f_i)$ 
14: end for

```

---

**Definition: Reverse Set of Influential Node.** Given a graph  $G(V, E)$  and a node  $v$ , the RIN query retrieves all the nodes  $v \in V$  that have  $u$  as their highest influential recommender node.

Consider a social network with five different friends where the edge influence between each pair of friends is calculated. Figure 3 shows an example of the reverse set of influential friends for each friend in the social network.



**Figure 3.** Example of a reverse set of influential nodes.

Considerable attention should be given to the reverse influential recommendation due to its importance in different domains, such as decision support applications, location recommendation applications, profile-based marketing, and others.

### 3.2. FoF-SocialI: Friend-of-Friend Influence-Based Location Recommendation Algorithm

While analyzing social network graphs, we surprisingly found that the potential a friend might have to get a more precise and accurate location recommendation from a friend of friend could be higher than that computed from the directly connected friends. Since TBA considers only direct friends, a new algorithmic framework FoF-SocialI is proposed that enhances location recommendation by considering different levels of friendship. Figure 4 shows the FoF-SocialI framework.



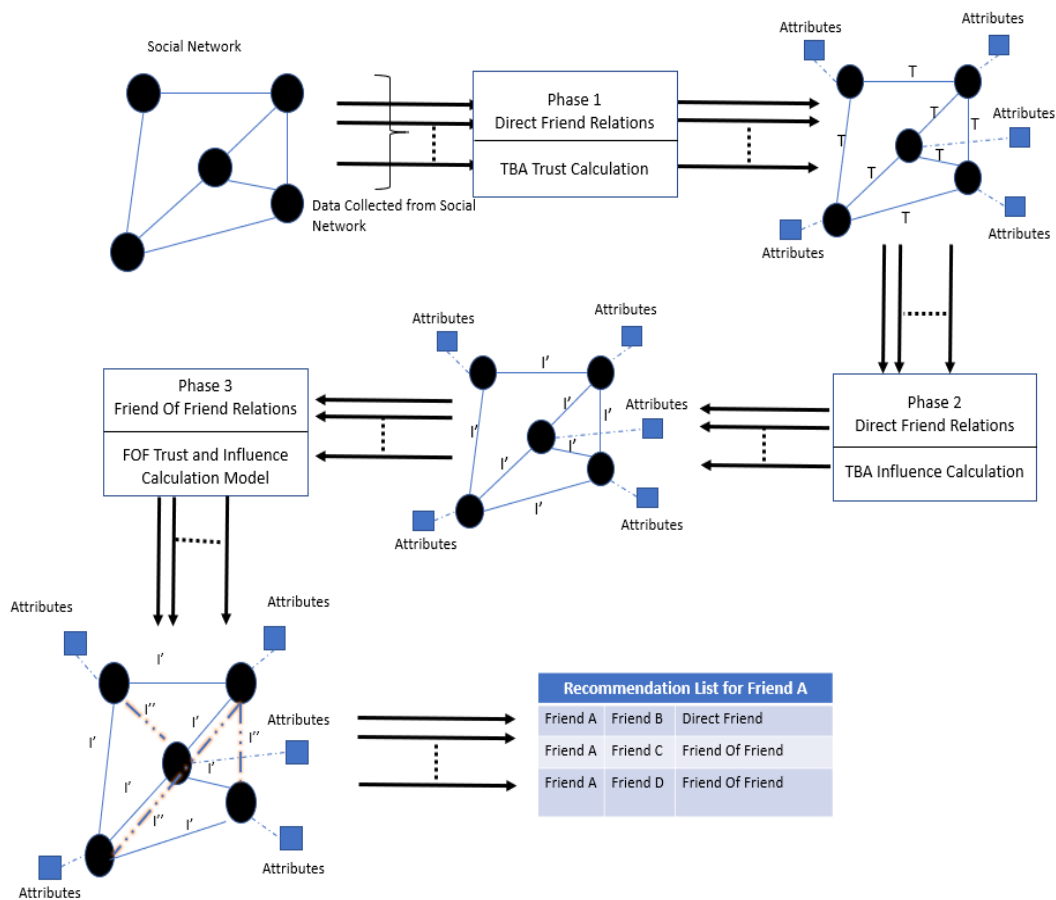


Figure 4. Friend-of-Friend social influence framework.

One of the main challenges that we address in building FoF-SocialII is dealing with the colossal size of social networks. Therefore, we constructed subgraphs based on the concept of ego social networks [31] according to the current context of the querying user requesting recommendation. Ego networks consist of an ego node, and all nodes to whom the ego has a connection at a certain path length are added. We determined the path length in ego networks using a connectivity degree (CD) value, which determines the depth of the friendship relations. Figure 5 shows an example of an ego social network with different degrees of connectivity.

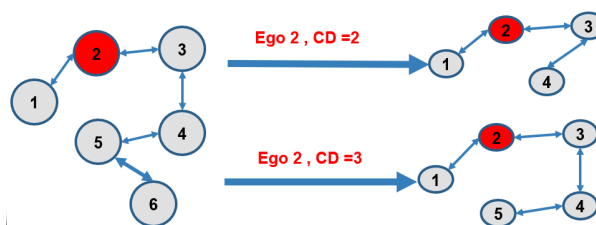


Figure 5. Ego social network.

Then, the actual recommendation is performed using the proposed FoF-SocialII algorithm. The idea behind the algorithm is building virtual relations between Friends of Friend, and these virtual relations are determined based on the specified connectivity degree between friends resulting in a new social network with more connected friends, as shown in Figure 6. We then apply the trust-based algorithm on the newly created social network to calculate the influences based on the virtually connected augmented social network.

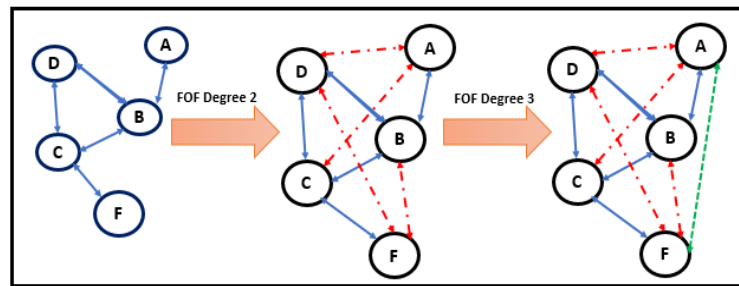


Figure 6. Newly generated social network with added virtual relations.

The FoF-SocialI algorithm describes the key steps for creating a new social network based on virtual relations as given in Algorithm 2. First, a social graph is constructed and then clustered into several subgraphs using the concept of ego social networks. In ego networks, an ego node is chosen, and all nodes to whom ego has a connection at a certain path length are added. We determine the path length in ego networks using a connectivity degree (CD) value, which determines the depth of the friendship relations (lines 1–3). In (lines 4–10), the new graph with virtual relations is created by binding the new friends of a friend to a particular friend creating a new network with more connected friends.

---

**Algorithm 2:** Friend-of-Friend (FoF-SocialI) Algorithm: Formation of new social network based on virtual relations

---

```

1: Initiate Graph<V,E>
2: CD ← 2
3: New_Graph ← (make_ego_graph(Graph, CD)) // make_ego_graph is a built-in function in R Programming Language
4: Virtual_Graph ← New_Graph
5: for each friend  $v_i$  in  $i := 1$  to size(New_Graph) Do
6:   for each friend  $v_j$  in  $j := i$  to size(New_Graph) Do
7:     if New_Graph [ $v_i,2$ ] = New_Graph [ $v_j,1$ ] then
8:       Virtual_Graph + 1 ← New_Graph[[ $v_i,1$ ],[ $v_j,2$ ]]
9:   end for
10: end for

```

---

With the new social graph, we apply the proposed trust-based algorithm (TBA) to compute the trust and influence between the newly connected friends to provide location recommendation services.

#### 4. Experimental Results

To test and validate the proposed algorithm, we combined both experimental and modeling approaches in a segregated approach, as described below. For our evaluation, we employed three real location-based social networks datasets, namely Brightkite [32], Gowalla [32], and Weeplaces [33], which are available for public use. The Weeplaces dataset has 7,658,368 check-ins from 16,091 different users over 971,309 different locations and 119,930 friendship relations. The Brightkite dataset has 4,491,143 check-ins from 58,228 users over 772,933 locations and 214,078 friendship relations. The Gowalla dataset has 6,442,890 check-ins from 196,591 users over 1,280,956 locations and 950,327 friendship relations.

On examining these datasets, Brightkite and Gowalla had several problems that can influence the recommendation process:

1. Although a considerable number of friends have different friendship relations, they do not have any saved data regarding their previously visited locations.
2. Although some friends have check-in data stored in the dataset, no information regarding their friendship relations is available.
3. Demographic information is not available for any of the friends in the dataset.

Handling these problems was done in two main phases: data filtration and data augmentation.

**Data Filtration.** In this phase, we eliminated all friends who did not have any stored information about their previously visited locations. Secondly, we removed all friends who did not have any friendship relations in the network. The outcome of the data filtration phase is a strongly connected network. The newly synthetic datasets are shown in Figure 7.

**Data Augmentation.** The Gowalla, Weeplaces, and Brightkite datasets have no demographic information to define the friends’ characteristics. The difference in friends’ characteristics can have a significant role in predicting the behaviors and interests of each user, which in turn affects the recommendation process. After observing the influence of a set of attributes in the recommendation process, we augmented the most effective ones to the dataset, which in turn has a significant effect in the recommendation process.

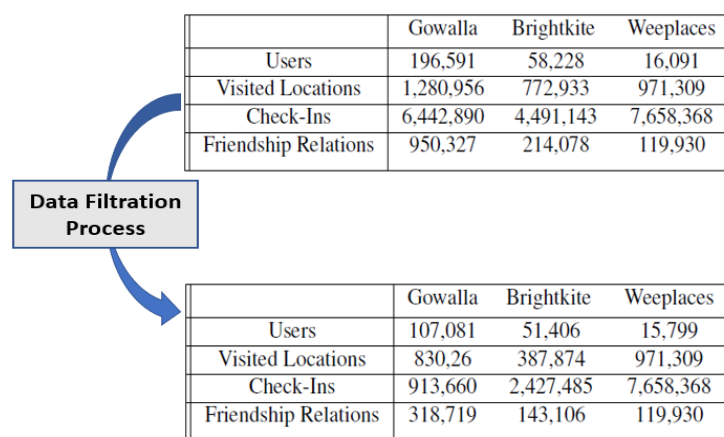


Figure 7. Gowalla, Brightkite, and Weeplaces updated datasets.

On analyzing these datasets, some necessary statistical measures were considered. It was observed that Gowalla and Weeplaces share the same structural behavior, where all nodes of both networks are highly connected, forming one big cluster; on the contrary, Brightkite is composed of 547 clusters. In addition, the average degree of each social network, which measures the number of friends each user has in the network is very low, between 4 and 16. Furthermore, in terms of the clustering coefficient, which computes the extent to which nodes of the network tend to make clusters, the highest tendency was for the Brightkite dataset due to the weak connectivity degree between its nodes. We also observed the assortativity coefficient of the network, which determines the preference of users to attach to other users, based on the node degree in our case. The diameter of the graph is another computed measure to find the longest and shortest path in the network; this is used to measure the friend-of-friend influence. Table 2 summarizes the statistical measure of these datasets.

Table 2. Statistical measures for different datasets.

Statistical Measure	Dataset		
	Gowalla	Brightkite	Weeplaces
Connectivity Strength	Strong	Weak	Strong
# Clusters	1	547	1
Average Degree	9.66	7.35	15.18
Assortivity Degree	−0.029	0.01	−0.042
Clustering Coefficient	0.023	0.11	0.053
Diameter	14	16	14

In this work, the first set of experiments was performed to verify the effectiveness of trust-based algorithm (TBA) in detecting the influential nodes compared to other prior algorithms.

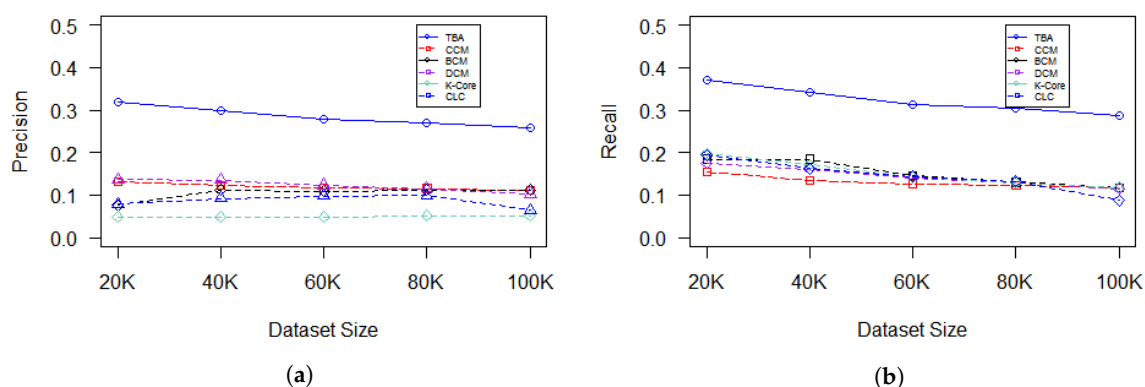
TBA performance was compared to both neighborhood-based centrality measures, Degree Centrality Measure (DCM) [1], K-Core decomposition [2] and Local Centrality with a Coefficient (CLC) [7], and path-based centrality measures, Closeness Centrality Measure (CCM) [1] and Betweenness Centrality Measure (BCM) [1]. Because a social network is modeled as a graph in the rest of this paper, we use both terms network and graph interchangeably to identify the same concept, namely, the social network. We evaluated the efficiency of TBA using the Gowalla, Brightkite, and Weeplaces datasets. The nodes' influences were measured using DCM, BCM, CCM, K-Core decomposition, CLC, and TBA, respectively. TBA achieves a better performance in detecting vital nodes across different datasets' structure compared to other algorithms. Furthermore, in Table 3, we show the CPU (Central Processing Unit, Core i5 2.3 GHz) time for DCM, BCM, CCM, K-Core decomposition, CLC, and TBA on the three real networks Brightkite, Gowalla, and Weeplaces. We can see that our algorithm requires slightly more time than DCM, BCM, CCM, and K-Core decomposition but far less time than CLC.

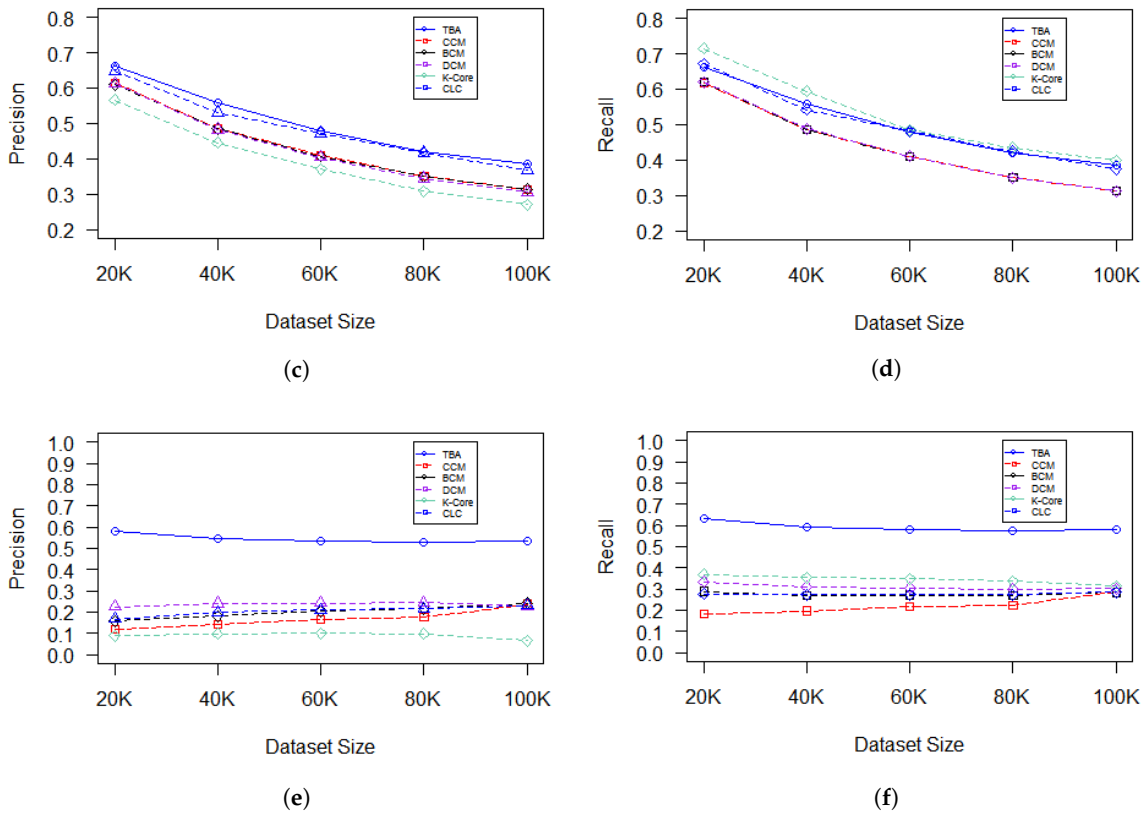
**Table 3.** The CPU time of six measures on three real networks.

Dataset	DCM/s	BCM/s	CCM/s	K-Core/s	CLC/min	TBA/s
Gowalla	0.0730	0.1060	0.0880	0.1780	> 10.0	4.6022
Weeplaces	0.0650	0.1530	0.1100	0.2610	> 10.0	6.7243
Brightkite	0.1560	0.4390	1.6701	0.2490	2.03	4.6252

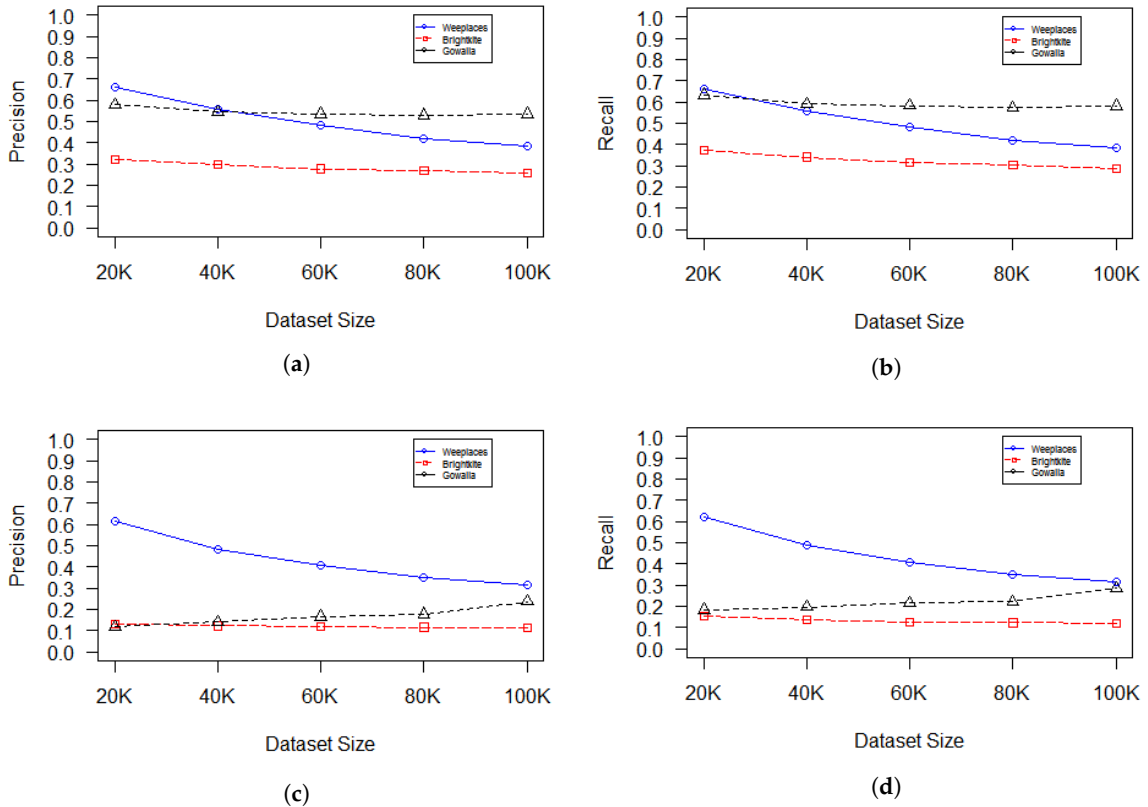
Although the performance of all algorithms in detecting influential nodes is remarkably low in weakly connected graphs due to the clustering behavior of the graph, TBA still shows a better performance compared to the other algorithms, which indicates the robustness of TBA. The experimental results are shown in Figure 8. Although the precision and recall values decrease with the increase of the dataset size, it is noticeable that, with some algorithms, such as CCM, the precision and recall values increase; this is due to the characteristics of the CCM algorithm. CCM searches for the node that is closest to all other nodes in the network. Thus, when the dataset size increases, the tendency of a node to be more central to other connected nodes also increases, therefore the possibility to diffuse information to more nodes also increases. In addition, it is noticeable that there is a big gap between the precision and recall of different dataset groups. Since Gowalla and Weeplaces share some essential characteristics, such as both datasets having a strong connectivity degree and consisting of only one cluster, the possibility to influence more nodes in the network is much higher. On the other hand, Brightkite is considered a weakly connected dataset with 547 clusters; thus, information diffusion cannot be guaranteed to all nodes in the network.

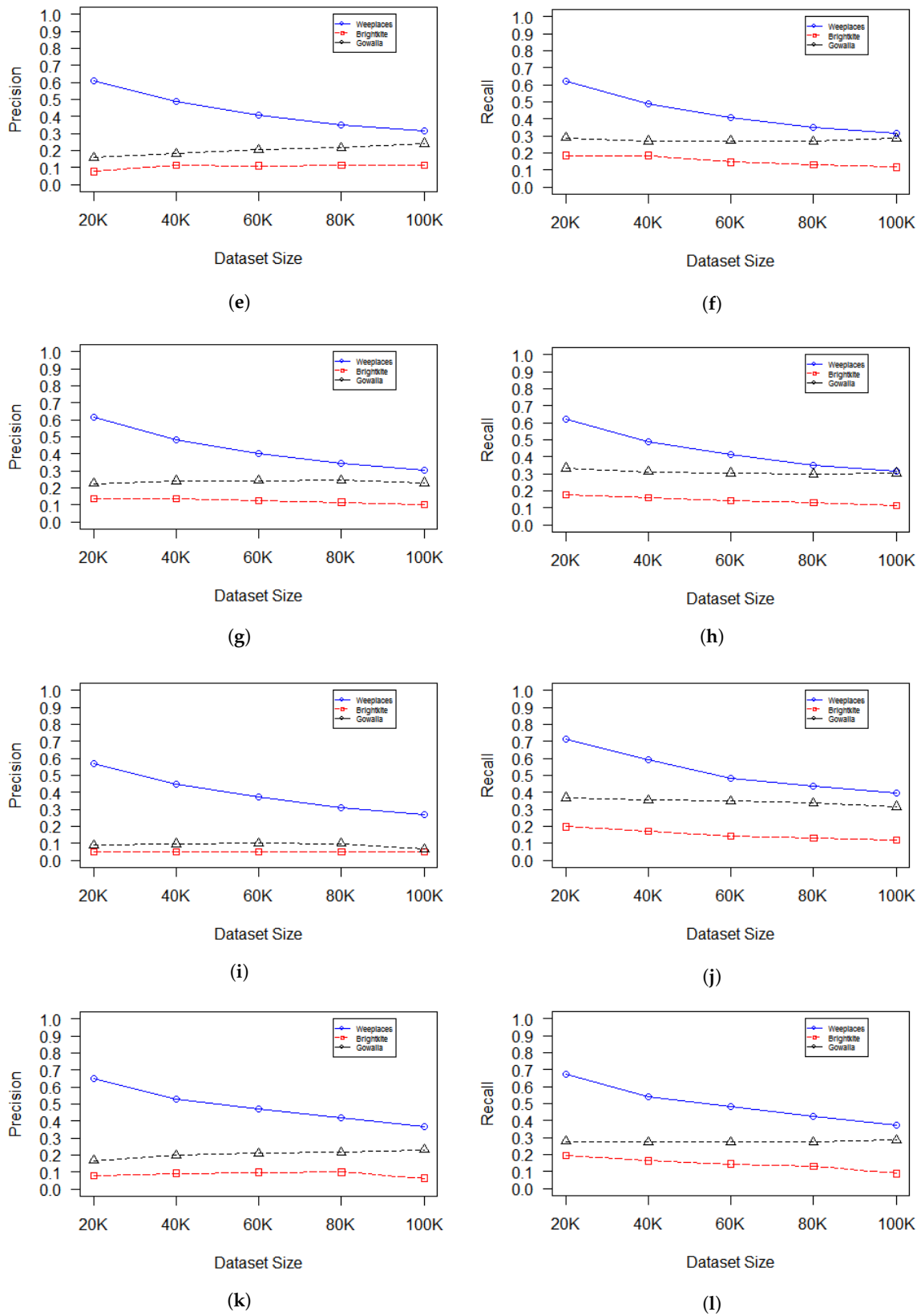
In Figure 9, the performance of each centrality measuring algorithm with different structural forms of datasets is examined. The experiments show that the performance of any centrality measure algorithm increases with datasets that have a higher connectivity strength, a higher average degree, and a lower diameter. Furthermore, the proposed algorithm outperforms other mentioned algorithms with datasets that have a higher connectivity strength and lower average degree for its nodes.





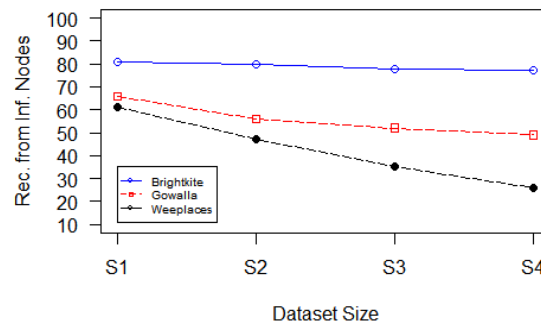
**Figure 8.** Performance comparison in terms of precision and recall for each dataset. (a) BrightKite precision; (b) BrightKite recall; (c) Weeplaces precision; (d) Weeplaces recall; (e) Gowalla precision; (f) Gowalla recall.





**Figure 9.** Performance comparison in terms of precision and recall for each dataset. (a) trust-based algorithm (TBA) precision; (b) trust-based algorithm (TBA) recall; (c) closeness centrality measure (CCM) precision; (d) closeness centrality measure (CCM) recall; (e) betweenness centrality measure (BCM) precision; (f) betweenness centrality measure (BCM) recall; (g) degree centrality measure (DCM) precision; (h) degree centrality measure (DCM) recall; (i) K-Core precision; (j) K-Core recall; (k) local centrality with a coefficient (CLC) precision; (l) local centrality with a coefficient (CLC) recall.

The next experiment was performed to examine the impact of influential nodes on the recommendation process. The experiment shows that 79% of the recommendations come from influential nodes in graphs with a higher number of clusters (e.g., Brightkite). This percentage decreases until it reaches 42% in graphs composed of one cluster (e.g., Gowalla and Weepplaces), as shown in Figure 10. These results prove that the graph structure has a significant influence, despite the importance of vital nodes.



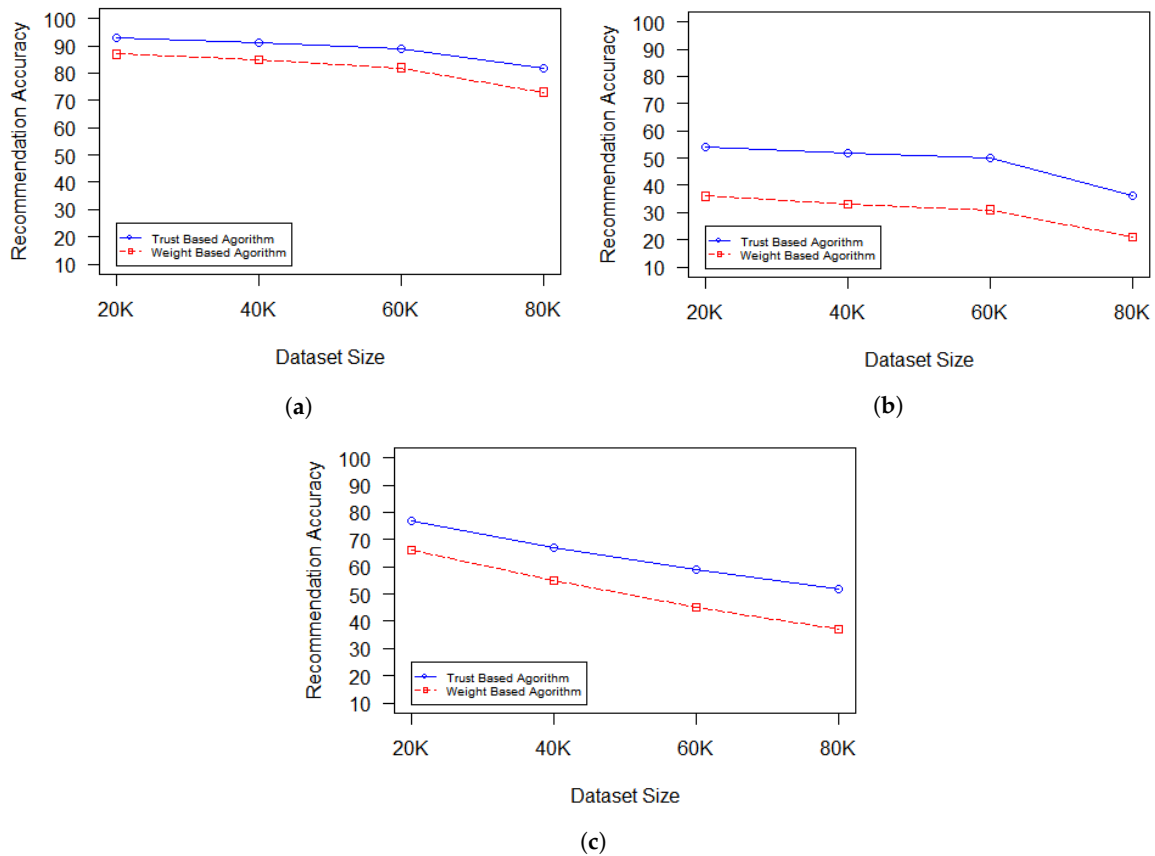
**Figure 10.** Recommendation from influential nodes.

Figure 11 shows a comparison between the trust-based algorithm and weight-based influence calculation algorithm proposed [16] to evaluate the recommendation capabilities of each algorithm. The evaluation proceeds as follows: firstly, the edge influence is computed between each friend and other socially related friends; secondly, a recommendation score list based on the computed edge influences is created; finally, the friend with the greatest edge influence is considered the most trusted reference for recommendation. The experiments were performed using the Gowalla, Brightkite, and Weepplaces datasets, and the results are impressive. In the three experiments, TBA outperforms algorithm [16] in the recommendation accuracy. In addition, it can be seen that, as the size of the dataset increases, TBA is still capable of maintaining an acceptable decrease in both the recommendation accuracy and the execution time, as shown in both Figures 11 and 12.

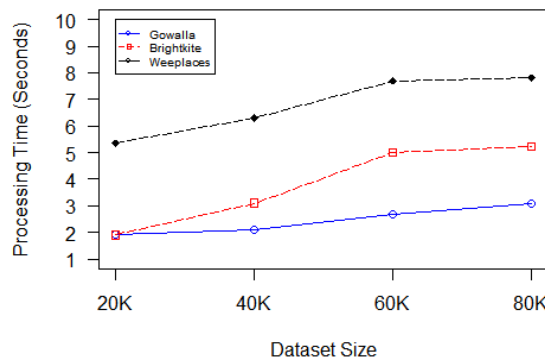
Through analyzing the obtained results, we observed the following:

1. TBA proves to have a higher efficiency on strongly connected graphs such as Gowalla and Weepplaces with approximately 70% precision compared to 35% precision in weakly connected graphs such as Brightkite;
2. As the average degree of the network increases, the recommendation accuracy using the proposed algorithm increases. Weepplaces shows the highest precision among all the datasets;
3. As the number of clusters of a graph increases, the tendency to diffuse information from vital nodes increases;
4. The TBA algorithm proves that recommendation does not always come from the most influential nodes in the network. The experiments show that almost 35% of the social network participants prefer to get their recommendation from direct friends rather than the network's biggest influencers;
5. The K-Core decomposition algorithm has the lowest in recommendation accuracy among all of the algorithms.

These findings summarize our previous discussion. In the remainder of this section, we present the results obtained from the experiments on the Friend-of-Friend algorithm.



**Figure 11.** Effect of dataset size on recommendation accuracy. (a) BrightKite dataset; (b) Gowalla dataset; (c) Weeplaces dataset.



**Figure 12.** Effect of dataset size on processing time.

Several experiments with different connectivity degrees were performed. It was observed that, as the degree of the friend of a friend increases, the influence of the new virtual friends decreases and the processing time increases. Figure 13 illustrates the results of experiments that were performed on three different friends with different connectivity degrees. The results show that friends with a connectivity degree of 2 enhance the recommendation location service by 46% followed by 35% for a connectivity degree of 3; this continues decreasing until it reaches 3% with a connectivity degree of 5.



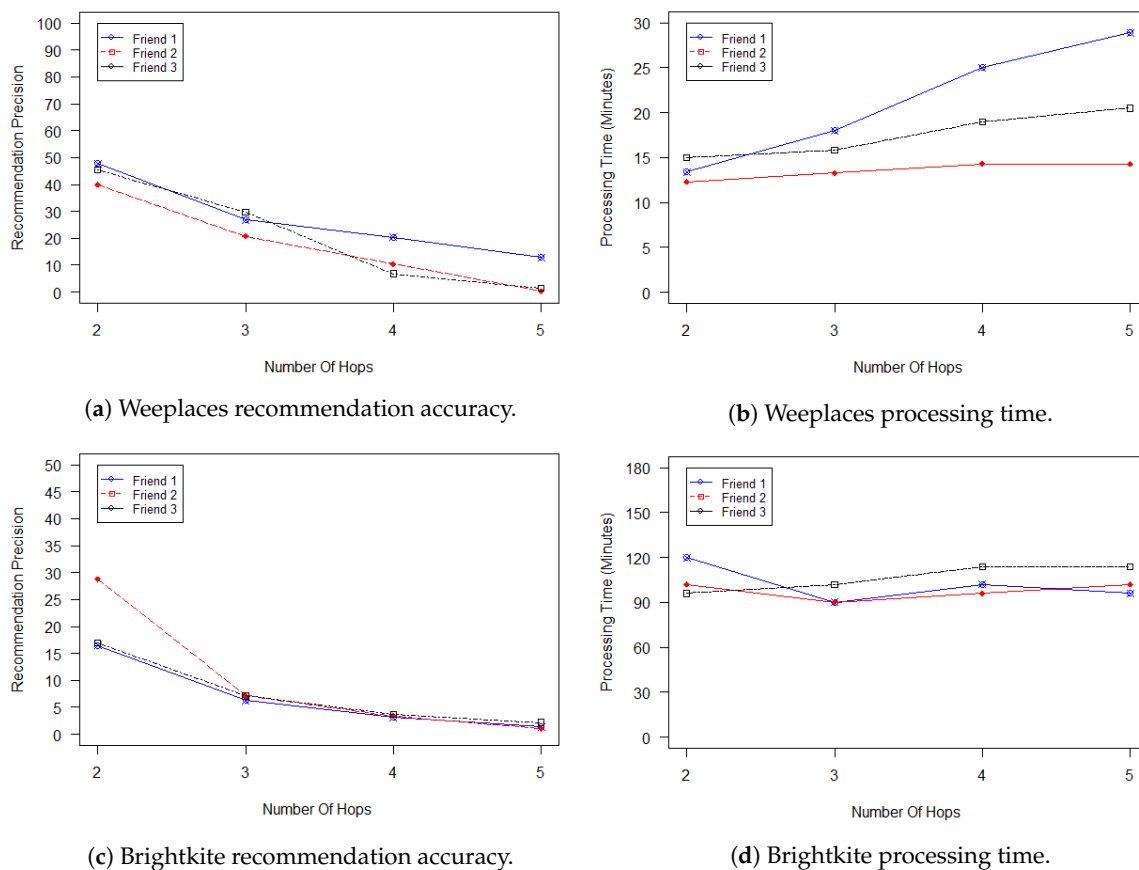


Figure 13. Friend of friend recommendation.

## 5. Conclusions

In this paper, we propose an efficient algorithm (TBA) for discovering influential nodes in online social networks and enhancing the location recommendation services. In particular, the introduced algorithm focuses on both the social and spatial information associated with each friend in the network. The experimental results show that the TBA algorithm outperforms the other five algorithms tested in different social networks, as TBA achieves 35% enhancement in identifying influential nodes on strongly connected networks and approximately 20% on weakly connected networks. Furthermore, the TBA algorithm proves its efficiency in providing location recommendation services when compared to other prior algorithms. Additionally, we propose a novel Friend-of-Friend recommendation (FoF-SocialII) algorithm. FoF-SocialII utilizes the concept of ego networks in clustering the social network followed by building imaginary relations between a friend and his/her indirect friends to compute the new social influence acquired from the new friends. Experimentally, FoF-SocialII proves that, in almost 50% of the cases, people are willing to receive their recommendation from their indirect friends rather than from directly tied friends. The efficiency of the proposed algorithms was verified using three online social networks. Although the experimental results prove that the proposed algorithms surpass relevant former algorithms for all of the datasets, the size of social networks continues to increase remarkably; therefore, introducing efficient algorithms to detect influential nodes is a long-term challenge.

With few minor modifications, the proposed algorithms can be adopted in other types of recommendation systems, such as e-commerce applications and tourism recommendation systems, amongst others. There are some future improvements to be considered to enhance the performance of the proposed algorithms. For example, developing a distributed version of both the TBA algorithm and FoF-SocialII algorithm is an interesting extension to this work. In addition, integrating more domain knowledge to the datasets could significantly improve the modeling of

temporal influence. Additionally, since most of the work done only considered the network structure with its geospatial information, disregarding the influence of semantics underlying these datasets, the proposed algorithms could also be enhanced by introducing the semantics that define the datasets. Furthermore, most of the presented algorithms, including those introduced in this paper, do not consider the importance of community detection. People tend to form communities with those who share the same choices and preferences, and the presence of communities remarkably affects the process of detecting influential nodes. It would be interesting to improve the proposed algorithms to utilize the presence of communities.

**Author Contributions:** H.M.W. and H.M.O.M. analyzed the data; H.M.W. and S.S.G. designed and performed the experiments; and H.M.W. and H.M.O.M. wrote the paper.

**Funding:** This work was supported by a fund from The British University in Egypt.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Landherr, A.; Friedl, B.; Heidemann, J. A Critical Review of Centrality Measures in Social Networks. *Bus. Inf. Syst. Eng.* **2010**, *2*, 371–385.
2. Alvarez-Hamelin, J.I.; Dall'Asta, L.; Barrat, A.; Vespignani, A. Large Scale Networks Fingerprinting and Visualization Using the K-core Decomposition. In Proceedings of the 18th International Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 5–8 December 2005; pp. 41–50.
3. Ver Steeg, G.; Galstyan, A. Information Transfer in Social Media. In Proceedings of the 21st International Conference on World Wide Web, Lyon, France, 16–20 April 2012; pp. 509–518.
4. Yang, J.; Leskovec, J. Modeling Information Diffusion in Implicit Networks. In Proceedings of the 2010 IEEE International Conference on Data Mining, Sydney, Australia, 13–17 December 2010; pp. 599–608.
5. Ye, S.; Wu, S.F. Measuring Message Propagation and Social Influence on Twitter.Com. In Proceedings of the Second International Conference on Social Informatics, Laxenburg, Austria, 27–29 October 2010; pp. 216–231.
6. Bonchi, F. Influence Propagation in Social Networks: A Data Mining Perspective. In Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, Lyon, France, 22–27 August 2011; p. 2.
7. Zhao, X.; Liu, F.; Wang, J.; Li, T. Evaluating Influential Nodes in Social Networks by Local Centrality with a Coefficient. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 35.
8. Margaris, D.; Vassilakis, C.; Georgiadis, P. Recommendation information diffusion in social networks considering user influence and semantics. *Soc. Netw. Anal. Min.* **2016**, *6*, 1–22. [[CrossRef](#)]
9. Margaris, D.; Vassilakis, C.; Georgiadis, P. Knowledge-Based Leisure Time Recommendations in Social Networks. In *Current Trends on Knowledge-Based Systems: Theory and Applications*; Springer: Cham, Switzerland, 2017; pp. 23–48.
10. Meo, P.D.; Musial-Gabrys, K.; Rosaci, D.; Sarnè, G.M.L.; Aroyo, L. Using Centrality Measures to Predict Helpfulness-Based Reputation in Trust Networks. *ACM Trans. Internet Technol.* **2017**, *17*, 8. [[CrossRef](#)]
11. Peng, S.; Yang, A.; Cao, L.; Yu, S.; Xie, D. Social influence modeling using information theory in mobile social networks. *Inf. Sci.* **2017**, *379*, 146–159. [[CrossRef](#)]
12. Adali, S.; Escriva, R.; Goldberg, M.K.; Hayvanovych, M.; Magdon-Ismail, M.; Szymanski, B.K.; Wallace, W.A.; Williams, G. Measuring behavioral trust in social networks. In Proceedings of the IEEE International Conference on Intelligence and Security Informatics, Vancouver, BC, Canada, 23–26 May 2010; pp. 150–152.
13. Song, H.; Pei, Q.; Xiao, Y.; Li, Z.; Wag, Y. A Novel Recommendation Model Based on Trust Relations and Item Ratings in Social Networks. In Proceedings of the 2017 International Conference on Networking and Network Applications (NaNA), Kathmandu, Nepal, 16–19 October 2017; pp. 17–23.
14. Bao, J.; Zheng, Y.; Mokbel, M.F. Location-based and Preference-aware Recommendation Using Sparse Geo-social Networking Data. In Proceedings of the 20th International Conference on Advances in Geographic Information Systems, Redondo Beach, CA, USA, 6–9 November 2012; pp. 199–208.
15. Hakan, B.; Pinar, K. Context-Aware Friend Recommendation for Location Based Social Networks Using Random Walk. In Proceedings of the 25th International Conference Companion on World Wide Web, Montreal, QC, Canada, 11–15 April 2016; pp. 531–536.

16. Hangal, S.; MacLean, D.; Lam, M.S.; Heer, J. All Friends are not Equal: Using Weights in Social Graphs to Improve Search. In Proceedings of the Workshop on Social Network Mining & Analysis, ACM SIGKDD, Washington, DC, USA, 25–28 July 2010; pp. 142–149.
17. Wagih, H.M.; Mokhtar, H.M.O.; Ghoniemy, S.S. Location Recommendation Based on Social Trust. In Proceedings of the 13th International Conference on Semantics, Knowledge and Grids (SKG), Beijing, China, 14–15 August 2017; pp. 50–55.
18. Kim, H.; Beznosov, K.; Yoneki, E. Finding Influential Neighbors to Maximize Information Diffusion in Twitter. In Proceedings of the 23rd International Conference on World Wide Web, Seoul, Korea, 7–11 April 2014; pp. 701–706.
19. Zhang, X.; Zhu, J.; Wang, Q.; Zhao, H. Identifying Influential Nodes in Complex Networks with Community Structure. *Knowl.-Based Syst.* **2013**, *42*, 74–84. [[CrossRef](#)]
20. Narayanam, R.; Narahari, R. A Shapley Value-Based Approach to Discover Influential Nodes in Social Networks. *IEEE Trans. Autom. Sci. Eng.* **2011**, *8*, 130–147. [[CrossRef](#)]
21. Rossi, M.; Malliaros, F.D.; Vazirgiannis, M. Spread It Good, Spread It Fast: Identification of Influential Nodes in Social Networks. In Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 101–102.
22. Malliaros, F.D.; Rossi, M.G.; Vazirgiannis, M. Locating influential nodes in complex networks. *Sci. Rep.* **2016**, *6*, 19307. [[CrossRef](#)] [[PubMed](#)]
23. Basuchowdhuri, P.; Majumder, S. Finding Influential Nodes in Social Networks Using Minimum k-Hop Dominating Set. In Proceedings of the First, International Conference on Applied Algorithms, Kolkata, India, 13–15 January 2014; pp. 137–151.
24. Mengtian, L.; Zhang, R.; Hu, R.; Yang, F.; Yao, Y.; Yuan, Y. Identifying and ranking influential spreaders in complex networks by combining a local-degree sum and the clustering coefficient. *Int. J. Modern Phys. B* **2018**, *32*, 1850118.
25. Goyal, A.; Bonchi, F.; Lakshmanan, L.V.S. Discovering Leaders from Community Actions. In Proceedings of the 17th ACM Conference on Information and Knowledge Management, Napa Valley, CA, USA, 26–30 October 2008; pp. 499–508.
26. Davoudi, A.; Chatterjee, M. Social trust model for rating prediction in recommender systems: Effects of similarity, centrality, and social ties. *Online Soc. Netw. Media* **2018**, *7*, 1–11. [[CrossRef](#)]
27. Yang, J.; Leskovec, J. Overlapping Community Detection at Scale: A Nonnegative Matrix Factorization Approach. In Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, Rome, Italy, 4–8 February 2013; pp. 587–596.
28. Wei, H.; Pan, Z.; Hu, G.; Zhang, L.; Yang, H.; Li, X.; Zhou, X. Identifying influential nodes based on network representation learning in complex networks. *PLoS ONE* **2018**, *13*, e0200091. [[CrossRef](#)] [[PubMed](#)]
29. Romualdo, P.; Alessandro, V. Epidemic spreading in scale-free networks. *Phys. Rev. Lett.* **2001**, *86*, 3200–3203.
30. Galton, F. Regression towards mediocrity in hereditary stature. *J. Anthropol. Insti. Great Br. Irel.* **1886**, *15*, 246–263. [[CrossRef](#)]
31. Epasto, A.; Lattanzi, S.; Mirrokni, V.; Sebe, I.O.; Tai, A.; Verma, S. Ego-net Community Mining Applied to Friend Suggestion. *Proc. VLDB Endow.* **2015**, *9*, 324–335. [[CrossRef](#)]
32. Leskovec, J.; Krevl, A. SNAP Datasets: Stanford Large Network Dataset Collection. Available online: <https://snap.stanford.edu/data> (accessed on 20 June 2014).
33. Liu, Y.; Wei, W.; Sun, A.; Miao, C. Exploiting Geographical Neighborhood Characteristics for Location Recommendation. In Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM 14), Shanghai, China, 3–7 November 2014; pp. 739–748.

