IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 2, 2019, accepted October 17, 2019, date of publication October 22, 2019, date of current version November 1, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2948948

CIDF: A Clustering-Based Interaction-Driven Friending Algorithm for the Next-Generation Social Networks

AADIL ALSHAMMARI[1] AND ABDELMOUNAAM REZGUI[2]

[1]Department of Computer Science and Engineering, New Mexico Institute of Mining and Technology, Socorro, NM 87801, USA
[2]School of Information Technology, Illinois State University, Normal, IL 61761, USA

Corresponding author: Aadil Alshammari (aalshamm@cs.nmt.edu)

ABSTRACT Online social networks, such as Facebook, have been massively growing over the past decade. Recommender algorithms are a key factor that contributes to the success of social networks. These algorithms, such as friendship recommendation algorithms, are used to suggest connections within social networks. Current friending algorithms are built to generate new friendship recommendations that are most likely to be accepted. Yet, most of them are *weak* connections as they do not lead to any interactions. Facebook is well known for its Friends-of-Friends approach which recommends familiar people. This approach has a higher acceptance rate but the strength of the connections, measured by interactions, is reportedly low. The accuracy of friending recommendations is, most of the time, measured by the *acceptance rate*. This metric, however, does not necessarily correlate with the *level of interaction*, i.e., how much friends do actually interact with each other. As a consequence, new metrics and friending algorithms are needed to grow the next generation of social networks in a meaningful way, i.e., in a way that actually leads to higher levels of social interactions instead of merely growing the number of edges. In this paper, we develop a novel approach to build friendship recommender algorithms for the next-generation social networks. We first investigate existing recommender systems and their limitations. We also highlight the side effects of generating easily accepted but *weak* connections between people. To overcome the limitations of current friending algorithms, we develop a clustering-based interaction-driven friendship recommender algorithm and show through extensive experiments that it does generate friendship recommendations that have a higher probability of leading to interactions between users than existing friending algorithms.

INDEX TERMS Clustering algorithms, Facebook, friending algorithms, recommender systems, social networks.

I. INTRODUCTION

Digital Information has been exchanged through the World Wide Web for several years. According to [1], 98% of all stored data in the world in 2015 were of digital form. This data can be reached by about 1 billion users who are connected to the Internet. IDC's ‘‘Digital Universe‘‘ reported that the digital data exchanged in the World Wide Web will reach 40 trillion gigabytes in 2020 [2]. The report estimated that there will be more than 5 thousands gigabytes for every person on Earth. Searching and finding relevant information within this gigantic amount of data is challenging [3], [4]. Therefore, recommendation algorithms were introduced to help overcome this challenge [4]–[7]. A particularly challenging type of recommender algorithms are friendship recommendation algorithms used in large scale social networks.

Friending algorithms are used to find and connect people. Facebook, for example, is well known for its friending algorithm which recommends friendship connections [8], [9]. The approach Facebook uses is a Friends-of-Friends *(FoF)* algorithm which is designed to recommend connections between already-known people. Facebook recommends billions of friendship connections among its 2.41 billion monthly active users [9], [10].

Interaction is an essential part of relationships and it is one of the main reasons behind seeking online friendships [11]. Since the Facebook approach recommends already-known people, the acceptance rate of its FoF algorithm is high.

The associate editor coordinating the review of this manuscript and approving it for publication was Massimo Cafaro.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see http://creativecommons.org/licenses/by/4.0/

TABLE 1. Interaction rate.

Users interacting with friends using comments	2.45%
Users interacting with friends using likes	13.23%

However, most of the connections recommended by Facebook' FoF do not lead to any interactions. Based on our collected dataset explained in Section III-B, Table 1 shows that the interaction rate amongst declared friends is considerably low. Simply put, current friendship recommender algorithms are focused more on increasing the number of connections and less on the quality of the connections.

The key motivation behind our research is to develop a novel friendship recommendation algorithm that can be used to generate a new type of social networks where interactions are exchanged at a substantially higher rate than in current social networks. Our research has direct social and business impacts. A study was conducted by the UK Ministry of Housing, Community and Local Government emphasized that interactions are proven to have positive physical as well as mental health impacts on people. While our research would contribute in improving people's social life by promoting interactive relationships, it would also contribute particularly in reducing the phenomenon of social isolation that has become increasingly challenging among certain segments of the population, e.g., the elderly. In [12], [13] the authors expressed the view that, as people age, they are more likely to struggle with being socially isolated which can have negative impacts on their mental and physical health. In addition, this research would also contribute in increasing the business profitability of social networks. Clemons [14] emphasized that the *4 Ps* including *Participatory* are essential to "ensure traffic". The author argues that active participation and interactions are one of the keys for the success of a social network business.

Paper Organization

This paper is organized as follows. In the next section, we overview some related work about the most well-known recommender systems with more focus on friending algorithms. In Section 3, we explain our data collection process. In particular, we describe the real social networks data that we fetched to test our proposed friending approach. We also present a brief analysis of our dataset. In Sections 4 and 5, we present our interaction-driven friending (IDF) approaches and demonstrate how they generate better friendship recommendations than Facebook's FoF approach. In Section VI, we present our clustering-based IDF algorithm. Finally, in Section 7, we present the results of our experiments on real datasets that illustrate how our algorithm compares to previous ones.

II. RELATED WORK

A. RECOMMENDER SYSTEMS

Collaborative filtering and content-based recommender algorithms are two of the most common types of recommendation algorithms. Collaborative filtering has been successfully used in many applications to recommend products and services [6], [15]. Unlike content-based algorithms, a collaborative filtering approach is not totally dependent on the "central" user's profile. The algorithm also computes other users' behaviors to create a *nearest-neighbors* circle of users with similar behaviors to the central user [6], [16], [17]. Further behaviors of the computed neighbors play a main role in computing new recommendations to the central user. For example, Amazon uses a similar approach for their famous suggestion "Customers who bought this item also bought [18].

TABLE 2. Results of the experiment.

Algorithms	Connections rated as Known	Connections rated as Bad
CM	22.5% (mostly good)	50.5% (mostly unknown)
CPL	36.2% (mostly good)	43.3% (mostly unknown)
FoF	60.6% (mostly good)	20.7% (mostly unknown)
SONAR	85.9% (mostly good)	17.6% (7.6% unknown)

Content-based recommendation algorithms are dependent on users' content, such as their profile, preferences, interests and likes/dislikes, as the main driver to compute recommendations [19], [20].

The success of collaborative filtering and content-based algorithms in suggesting items did not expand to recommending friendships between users. An extensive study [19] was conducted on testing the effectiveness of recommender algorithms in suggesting friendship connections. The study tested the following approaches:

1) Content-Matching (CM)
2) Content-Plus-Link (CPL)
3) Friends-of Friends (FoF) and
4) SONAR algorithms

CPL is a content-filtering algorithm that discloses the social link path between the *target*[1] user and the recommended user. This is done to justify the decision of the recommendation. The SONAR algorithm is built with the same intention as the FoF algorithm which is to find already known people but with intense input. This is because SONAR uses public databases, i.e., publication databases, patent databases, organizational charts, etc. This study dissects recommendation algorithms and identifies their capabilities in recommending friendship connections. Table 2 summarizes the experimental results of the study.

The study concluded that algorithms built to find and suggest relationships between already known people result in a considerably higher acceptance rate than other approaches. The key behind this success is to *not* recommend strangers to people. Although both content-based filtering algorithms

[1] The *target* is a user for which a friending algorithm generates recommendations

(CM and CPL) were able to recommend friendships between users based on their similarity, they failed against the FoF approach in terms of acceptance rate.

B. INTERACTION PROBLEM IN EXISTING FRIENDING ALGORITHMS

Several studies reported the lack of interactivity amongst declared friends in Facebook. A study was conducted on a Facebook dataset of 4.2 million users which showed that users interact with an average of 15.1% [21]. In this study, this result is the best case scenario of the Facebook's FoF algorithm assuming that all of these interactions came from relationships recommended by the algorithm and *not* found by the users themselves.

Another study conducted on the interactions of Facebook's users reached the same conclusion about the lack of interactions between friends in Facebook [22]. The authors derived an *interaction graph* from a Facebook's social graph dataset by eliminating all non-interactive social connections. They found that the interaction graph is substantially smaller than the entire social graph. The study concluded that a recommended connection does not always translate into a meaningful (*interactive*) relationship and that *interactive* relationships are essential for the success of a social network.

To conclude, the FoF algorithm outperformed advanced filtering algorithms in terms of acceptance rate and it is a useful approach to find connections that will most likely be accepted. However, further filtering is needed to find *interactive* connections among Friends-of-Friends.

III. DATASET

We developed a web crawler that fetches publicly available users' profiles from Facebook.com. Only public data are fetched. For anonymity we replace users' IDs with random numeric IDs. Publicly available user profiles are not any different from private user profiles. Both profiles contain the same set of attributes. The only difference between the two is that one profile can be accessed by anyone while the other can only be accessed by the respective user's declared friends.

Every user's profile we collected has six different types of data:

1) User ID.
2) Gender.
3) Current city and hometown.
4) Self-reported interests such as movies, reading, etc.
5) Declared friends list.
6) Interactions.

The interactions data we collect are based on the latest posts fetched from the most recent 4-6 time-line pages. The number of posts we collected from each user's profile depends on the number of posts in each time-line page. The average of the number of posts collected is about 30 posts per user's profile.

Each post collected consists of the following:

- Post title.
- Post ID.
- IDs of users who commented on the post.
- IDs of users who liked the post.

Commenting or liking one's own post is NOT counted as an interaction with that post. To ensure that we collect an effective dataset, we ran our crawler in different regions of the US and the UK. As a result, we have accumulated 16624 user profiles in total.

In order to test our algorithms on a given user from the collected dataset, that user needs to have his/her friends' list public and all his/her declared friends' profiles are collected in our dataset. Therefore, we ran a simple code on the collected dataset to find and return all user IDs whose profiles are publicly available and whose declared friends' profiles exist in our collected dataset. This resulted in 25 subgraphs. Each subgraph contains the respective user and his/her friends. These 25 subgraphs contains a total of 10500 users.

A. ACCURACY METRIC & INTERACTIONS

In this research, a recommendation is accurate only if it leads to interactions. In our accuracy metric, an algorithm's accuracy of recommending interactive friendships (noted θ) is defined as follows:

$$\theta(Alg) = \frac{R_i}{R_{all}} \tag{1}$$

where:

- R_i is the total number of interactive friendships recommended by Alg.
- R_{all} is the total number of all friendships recommended by Alg.

We consider as an *interaction* within the social network (i.e., Facebook) any of the two following events:

1) commenting on a friend's post or
2) liking a friend's post.

A user is *interactive* with his/her friend if he/she commented on at least one of that friend's posts. A comment, intuitively, is a stronger type of interaction than a like. Therefore, when there is no comment in a given relationship, at least two "likes" must be made for that relationship to be considered interactive relationship. For example, as shown in Figure 2, user A is interactive with user x because one of the following conditions was met:

1) User A commented on at least ONE of user x's posts.
2) User A liked at least TWO of user x's posts.

B. DATASET STATISTICS

Our analysis of the collected dataset confirmed the findings of the papers mentioned above about the low interactions amongst Facebook's users. To calculate the average percentage of users who interacted with their friends, a key element was the friends size of each profile. Out of the 16624 profiles, 6551 users have their friends list private. Therefore, the dataset statistics presented in Figure 1 is based on the calculation over 10073 users.

As shown in Figure 1, the average percentage of users interacting with their friends is very low. Only an average of 2.45% of users *commented* on their friends' posts and

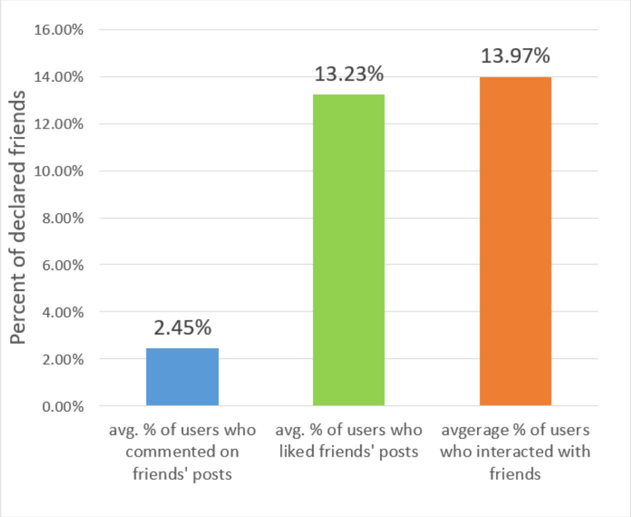


FIGURE 1. Dataset statistics.

TABLE 3. Posts categories.

Post Title	# of Posts	%	Categories
Updating Profile Picture	146,557	30.6%	60.41%
Updating Cover Photo	120,331	25.12%	
Adding Photos	22,468	4.69%	
Shared Link	10,546	2.2%	2.2%
General Posts	179036	37.38%	37.38%
Total	478,938	100%	100%

13.23% *liked* their friends' posts. Overall, the average percentage of interaction using likes or comments within our collected dataset is 13.93%. This means that about 86% of the declared friendship relationships are *weak*.

The dataset contains 478,938 posts created by 16,624 users. Our analysis of the dataset revealed that these posts can be categorized as shown in Table 3. Even though there are hundred of thousands of posts, the majority fall under one category which is posting personal photos. We acquire this information from posts titles. A post title can be one of the following:

1) "user name updated his/her profile picture".
2) "user name updated his/her cover photo".
3) Adding photos:
   a) "user name shared photos/a photo".
   b) "user name is with user name" posting photos/a photo.
   c) "album name" posting a photo album.
4) "user name shared a link".
5) "user name" a general post.

A general post is a photo post, a textual post or a status update. Therefore, the majority of posts are categorized as posting photos. The only type of post that can be interpreted into different types of interests is the fourth post title "user name shared a link". The links users share are about a variety of interests, e.g., comedies, politics, sports, etc. However, these posts are only 2.2% of the total number of posts.

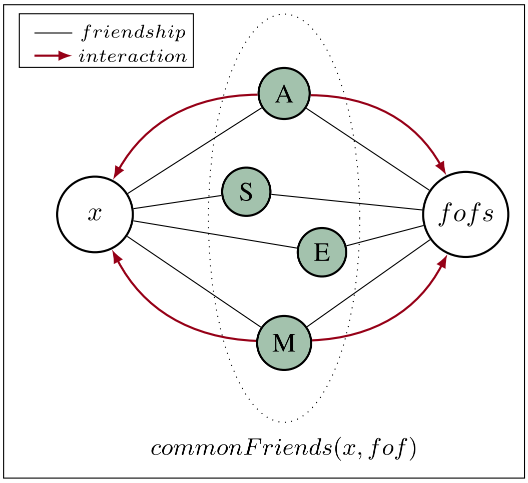


FIGURE 2. Sub-graph example for IDF-O.

IV. AN INTERACTION-DRIVEN FRIENDING ALGORITHM WITH OUTGOING EDGES: IDF-O

We have seen that the intention behind the *FoF* algorithm is to find already known people within a social graph. We have also seen that such intention is key to incentivize users to accept and, eventually, try a recommendation suggested by the recommender system. Therefore, when recommending friendships, *FoF* is an essential approach because it is built to avoid any user who might be considered as a stranger. During our development of friending algorithms, we ran a set of experiments on a content-based filtering (CBF) algorithm to filter FoFs and recommend *interactive* relationships. These experiments showed that users' self-reported interests are inconsistent with their real interests and that a CBF approach is not a valid solution to our research problem, i.e., the lack of interactivity amongst declared friends.

Within the context of our collected Facebook dataset, an interaction reflects an interest. An interaction with a post can either be a *like* or a *comment*. A *like* to a post is intuitively an interest. A *comment* on a post is also an interest even though one could argue that some comments can be negative. We randomly explored our dataset and did not find any negative comments. Of course, negative comments do exist but that was not the case in our scenario. This is because we only take into consideration interactions (i.e., *comments*) from friends, and negative *comments* do exist in a general context. Therefore, the interactions between friends are the actions towards their interests. Since we are calculating interactions as an indication of users' interests, the result is an up-to-date non-self-reported interests.

Studies in psychology showed that there is a positive correlation between the *closeness*[2] of friends and their similarities [23], [24]. These studies also hold true in social networks

[2] close friends: interactive friends who spend more time together than regular friends

because there is a correlation between people's behavior online and their behavior offline. In [25], the authors reviewed several papers that analyzed Facebook and concluded that online personalities reflect offline (real-life) personalities.

In our IDF-O approach (Figure 2), the similarities between the *target* and an FoF are determined by calculating the number of the *commonFriends* interacted with both x and *fof*. We discussed in Section III-B that categorizing the content (users' posts), in Facebook, was not possible due to the limited types of posts. Our IDF-O approach identifies the *persons* whom the *target* and FoF are interested in regardless of the content created by those persons. The algorithm recommends FoFs to the *target* only if there are similarities between them.

A. PROPOSED ALGORITHM

The pseudo-code of our IDF-O algorithm is given in Algorithm 1:

Algorithm 1 : IDF-O

```
 1: procedure IDF-O($x$, $G$)
 2:     for each vertex $f$ in $friends(x)$ do
 3:         for each vertex $ff \in friends(f)$ do
 4:             if $ff \notin friends(x)$ then
 5:                 $append(fofs, ff)$
 6:     for each vertex $fof$ in $fofs$ do
 7:         $commonFriends = friends(x) \cap friends(fof)$
 8:         $interactiveCounter = 0$
 9:         for each vertex $c$ in $commonFriends$ do
10:             if $c$ interacted with ($x$ and $fof$) then
11:                 $interactiveCounter += 1$
12:         if $counter < 4$ then
13:             $remove(fofs, fof)$
    return $fofs$
```

The IDF-O algorithm takes user x and the social graph as arguments. It starts by generating the set of x's friends-of-friends (*fofs*). Lines 3 and 4 ensure that the generated *fofs* list contains user x's friends-of-friends who are NOT already friends of x. Then, the algorithm considers every *fof* in the *fofs* set to determine whether the *fof* would have an interactive relationship with user x or not. This is done by the following four steps:

1) Generate the *commonFriends* of x and *fof* by calculating the intersection of their friends.
2) Iterate on every user c in the *commonFriends* set. If c interacted with both of x and *fof*, then *interactiveCounter* is increased by 1. This keeps track of the number of *commonFriends* whom x and *fof* interacted with.
3) If both x and *fof* interacted with a least 4 *commonFriends*, then *fof* remains in the *fofs* set. Otherwise, *fof* would not be considered as a possible interactive friend of x and, consequently, will be removed from the *fofs* set.
4) After iterating on the last *fof*, the algorithm will return the modified *fofs* set which only contains possible interactive friends-of-friends.

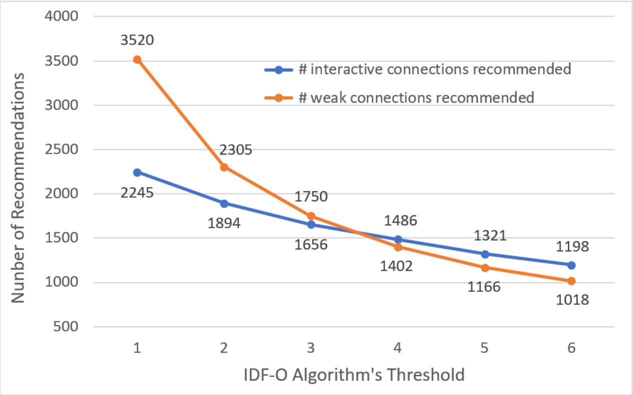


FIGURE 3. IDF-O threshold impacts.

The parameter "4" used in the algorithm is the algorithm's threshold which controls the intensity of the filtering process. This parameter is the least number of interactive *commonFriends* to qualify a friendship connection between x and *FoF*. In other words, it is the least number of similarities between x and *FoF* to approve a friendship recommendation between them. When this number (4) is increased, the accuracy of the algorithm ($\theta(Alg)$) increases but the total number of recommendations decreases. For example, lowering the parameter to 3 results in a higher number of recommendations with a lower $\theta(Alg)$ while increasing the parameter to 5 results in a lower number of recommendations with a higher $\theta(Alg)$. Figure 3, shows a graphical representation of the algorithm's results of recommending interactive and weak connections using different values of the threshold.

B. EXPERIMENTAL RESULTS

Both IDF-O and Facebook approaches are tested on the same set of users (10500). The valdiation methodology of the experiment is explained in Section VII-A. The IDF-O algorithm recommended 2888 recommendations. Out of the 2791 interactive relationships, the algorithm recommended 1486 interactive connections which is 53.24% of the total percentage of available interactive connections. This result can be improved if a higher number of users' *commonFriends* profiles were publicly accessible. The algorithm was able to recommend 76.42% of all interactive connections when the number of accessible *commonFriiends* was at least 30.

Overall, as shown in Figure 4, out of the 2888 connections recommended by our IDF-O algorithm, 1486 of them were of interactive relationships. This means that the accuracy of our IDF-O algorithm to recommend interactive connections (noted θ) is:

$$\theta(IDF-O) = \frac{1486}{2888} = 0.51 \qquad (2)$$

This accuracy of IDF-O is significantly higher than Facebook's FoF accuracy which is 0.26. The percentage

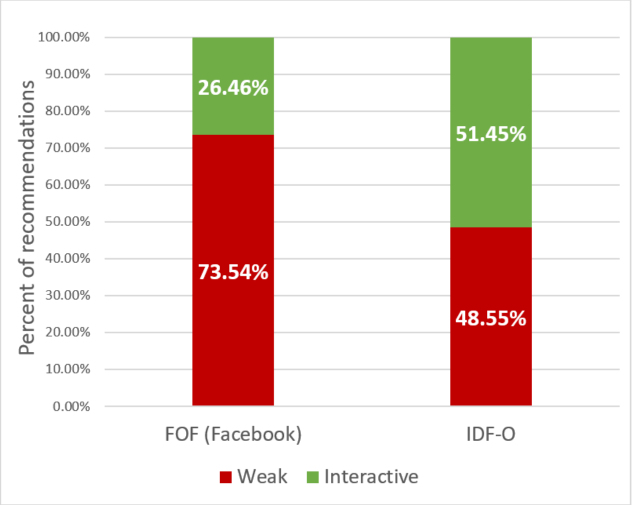


FIGURE 4. IDF-O vs. Facebook's FoF.

TABLE 4. Comparison summary.

Algorithm	# of Interactive Connections	θ	# of *weak* Connections	1-θ
FoF	2791	0.26	7758	0.74
IDF-I	1425	0.57	1086	0.43
IDF-O	1486	0.51	1402	0.49

of recommending *weak* relationships (1-θ) is reduced from Facebook's 0.74 to 0.49.

V. AN INTERACTION-DRIVEN FRIENDING ALGORITHM WITH INCOMING EDGES: IDF-I

We explained and evaluated our IDF-I approach in [26]. Simply put, the IDF-I algorithm determines the similarities between the *target* user and an FoF by calculating *incoming* interactions from x and fof to the same *commonFriends*. In [26], we tested the IDF-I algorithm on recommending interactive friends to a given user. In this paper, we re-evaluate our IDF-I algorithm with recommending interactive connections between x and fof, i.e., where interactions can come from x and/or fof. Table 4 summarizes the results of the experiments on IDF-I and IDF-O. The validation methodology of the experiments is explained in Section VII-A. The accuracy of a recommendation is measured by the likelihood of the recommended connection becoming an interactive relationship.

Although our IDF-I recommended a higher percentage of *interactive* connections than IDF-O, The *interactive* connections generated by IDF-O are not necessarily generated by IDF-I. There are *interactive* connections recommended by IDF-I which are not recommended by IDF-O and vice versa. IDF-I recommend fewer recommendations than IDF-O which is why IDF-O was able to find more *interactive* connections than IDF-I. However, the accuracy of IDF-I is higher than that of IDF-O because IDF-O recommends a considerably higher number of *weak* connections than IDF-I.

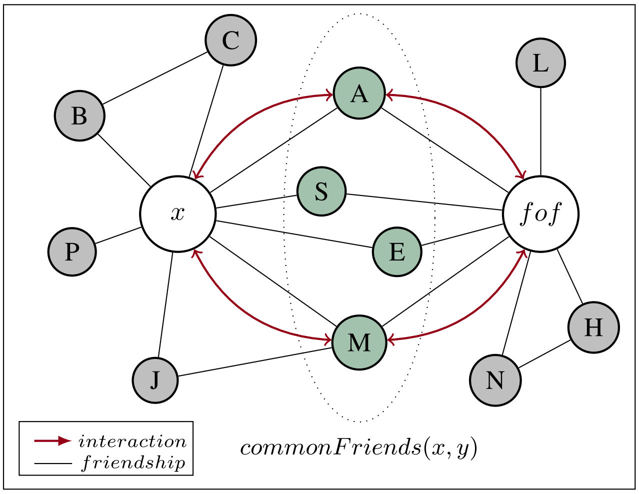


FIGURE 5. Sub-graph example for CIDF.

A possible approach to maximize the accuracy while maintaining the higher number of *interactive* connections recommended could be to combine the results of both algorithms. This, however, would create conflicting recommendations since an *interactive* connection generated by one algorithm might be generated as *weak* by the other algorithm. This leads us to our optimal algorithm where we use a k-means clustering approach to take advantage of both of our Interaction-Driven Friending approaches, IDF-I and IDF-O.

VI. A CLUSTERING-BASED INTERACTION-DRIVEN FRIENDING ALGORITHM: CIDF

In this section, we present our clustering-based interaction-driven friending (CIDF) algorithm (Algorithm 2). In this approach (Figure 5), *commonFriends* are the unidirectional *bridges* for interactions that help us identify an *interactive* connection between x and fof. First, we identify the *interactive commonFriends* of x and fof. Then, we take into consideration the intensity of interactions that are incoming to and outgoing from those *commonFriends*.

There are two advantages of CIDF over the two IDF approaches:

1) CIDF effectiveness is independent of the number of *commonFriends*. This means that CIDF is independent of the IDF algorithm's threshold. This is because CIDF calculates the intensity of interactions regardless of the size of *commonFriends* used for that calculation. The IDF approach calculates the number of *interactive commonFriends*.
2) CIDF combines both IDF approaches into one approach where incoming and outgoing interactions are calculated.

Every fof is plotted according to their (a, b) values where:

- a is the number of incoming interactions to *common Friends*(x, fof) from x and fof
- b is the number of outgoing interactions from *commonFriends*(x, fof) to x and fof

Then, using k-means, we cluster all *fofs* and the lowest interaction mean cluster contains *weak* connections which will not be recommended.

The algorithm's pseudo-code is given in Algorithm 2.

Algorithm 2 : CIDF

```
 1: procedure CIDF($x$, $G$)
 2:     $M = [\,]$
 3:     for each vertex $f$ in $friends(x)$ do
 4:         for each vertex $ff \in friends(f)$ do
 5:             if $ff \notin friends(x)$ then
 6:                 $append(fofs, ff)$
 7:     for each vertex $fof$ in $fofs$ do
 8:         $commonFriends = friends(x) \cap friends(fof)$
 9:         $I_{initiated} = 0$
10:         $I_{received} = 0$
11:         for each vertex $c$ in $commonFriends$ do
12:             if ($x$ and $fof$) interacted with $c$ then
13:                 $I$ = # of interactions from $x$ and $fof$ to $c$
14:                 $I_{initiated} += I$
15:             if $c$ interacted with ($x$ and $fof$) then
16:                 $R$ = # of interactions from $c$ to $x$ and $fof$
17:                 $I_{received} += R$
18:         $append(M, [fof, I_{initiated}, I_{received}])$
19:     $clusters = k\text{-}means(k, M)$
20:     $fofs = \{clusters - lowest\ mean\ cluster\}$
        return $fofs$
```

The initial steps (3 through 6) of our CIDF algorithm compute user x's *fofs* who are not declared friends with x. In this approach, we compute the total number of interactions that have been exchanged between *commonFriends* and both of the *target* user and the *fof*.

1) In line 2, we initiate a matrix M which will eventually contain the matrix input of the k-means clustering algorithm.
2) Then, we iterate on every *fof* in the set of *fofs*.
3) Within the *for* loop (in line 7), we start by calculating the *commonFriends* of users x and *fof* and we reset two variables $I_{initiated}$ and $I_{received}$.
4) Then, we iterate on every c in the calculated set of *commonFriends*.
5) Within this loop (in line 11), first we check if both x and *fof* interacted with c and we calculate the total number of these interactions and add them to $I_{initiated}$. This keeps track of the number of interactions that both x and *fof* initiated toward c (*incoming* interactions).
6) Second, we check if c interacted with both x and *fof* and we calculate the total number of these interactions and add them to $I_{received}$. This keeps track of the number of interactions that c initiated toward x and *fof* (*outgoing* interactions).
7) In line 18, at the end of the first iteration of the loop (starts in line 7), the algorithm would have calculated the first row of matrix M. The row consists of the *fof*, the total number of *incoming* interactions with *commonFriends* and the total number of *outgoing* interactions with *commonFriends*. Eventually, matrix M will consist of the calculated rows of all user x's *fofs*, as follows:

$$M = \begin{bmatrix} fof_1 & I^1_{initiated} & I^1_{received} \\ fof_2 & I^2_{initiated} & I^2_{received} \\ fof_3 & I^3_{initiated} & I^3_{received} \\ \cdots & \cdots & \cdots \\ fof_n & I^n_{initiated} & I^n_{received} \end{bmatrix}$$

This calculated matrix represents one case study out of 25 different case studies.
8) In line 19, the calculated M is passed with the number of clusters k to k-*means*().
9) In line 20, we delete every *fof* who is represented in the cluster with the lowest interactions mean. The clusters with higher interactions mean represent *fof* users who have a higher probability of forming *interactive* relationships with user x.
10) Lastly, our CIDF algorithm will return the modified set of *fofs*.

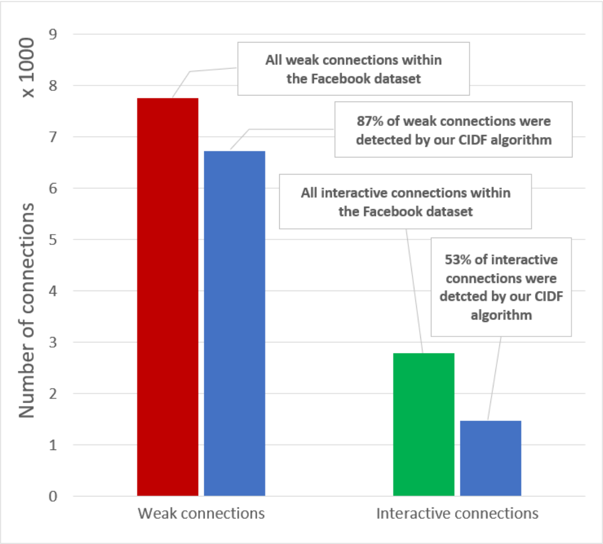
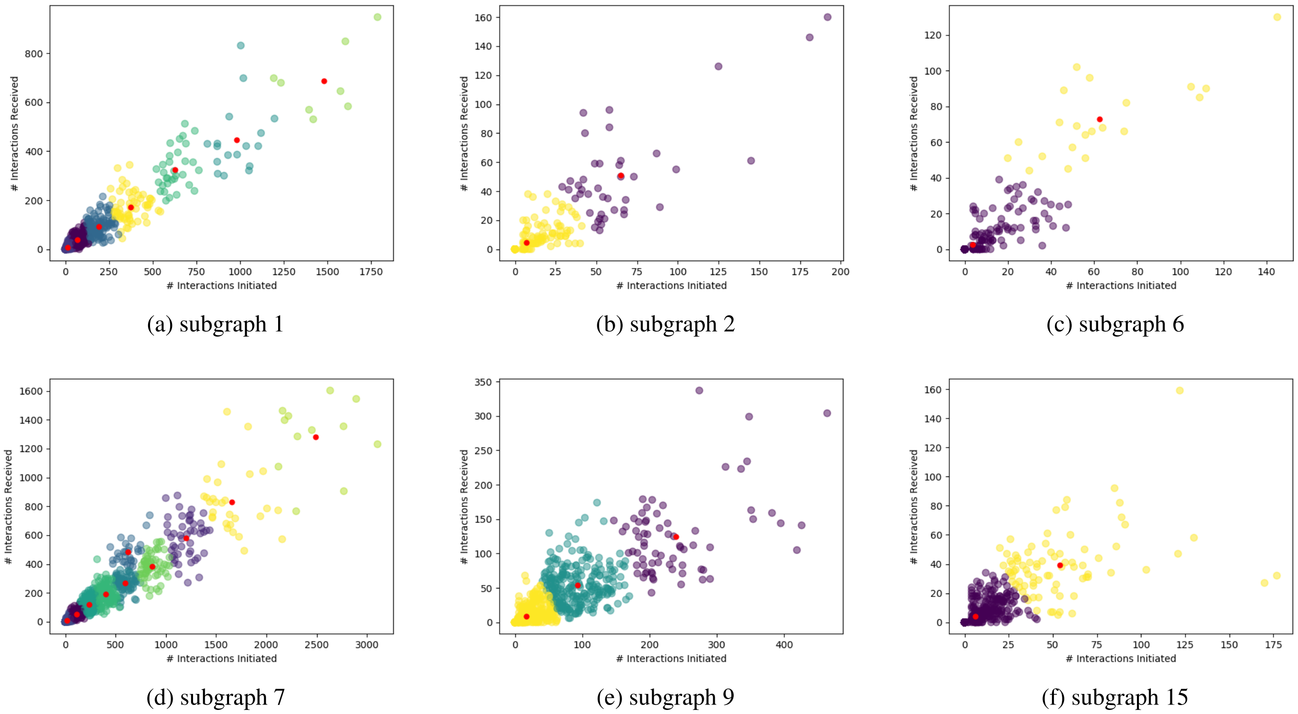


FIGURE 6. **CIDF effectiveness.**

The number of clusters (k) depends on the calculated matrix (M) of user x. Every user x is a special case. The levels of interactivity amongst user x's *fofs* is different from user to user. Having a fixed number of clusters is impractical. The co-ordinate of the centroid (the mean) of a cluster is based on the interaction intensity of the *fofs* in that cluster. The highest mean cluster holds *fofs* with the highest interactions mean which yields the highest probability of forming *interactive* connections. When the number of clusters is fixed with $k = 2$, the algorithm will result in the highest accuracy but the lowest number of recommended connections. This is because the higher mean cluster contains *fof* users who have considerably high levels of interactions with $commonFriends(x, fof)$.



(a) subgraph 1

(b) subgraph 2

(c) subgraph 6

(d) subgraph 7

(e) subgraph 9

(f) subgraph 15

FIGURE 7. Clusters plots examples from the CIDF experiments, more details in table 6.

TABLE 5. Comparison summary.

Algorithm	# of Interactive Connections	θ	# of *weak* Connections	1-θ
FoF	2791	0.26	7758	0.74
IDF-I	1425	0.57	1086	0.43
IDF-O	1486	0.51	1402	0.49
CIDF	1475	0.59	1022	0.41

However, the number of *interactive* friendships detected is still very low because the majority of *fofs* with a high interactivity rate, but still less than those in the high mean cluster, are in the ‘‘lower’’ mean cluster. When the number of clusters (k) is increased, more higher mean clusters are initiated and the mean of the lower mean cluster is decreased. This is because more of *fofs* with higher interaction intensity move away from the lowest mean cluster to higher mean clusters. Therefore, in our CIDF algorithm when the interaction intensity of the lowest cluster mean is still high, we recall the k-*means*() algorithm with a higher number of clusters ($k + 1$). The CIDF algorithm stops recalling k-*means*() when the interaction intensity of the lowest mean cluster is as low as 10.

VII. EXPERIMENT

Our collected Facebook dataset consists of 25 subgraphs with 10500 users’ profiles. Each subgraph has a ‘‘center’’ user (the *target* user) whose friends’ profiles all exist in our dataset. The algorithm takes the social graph (dataset) and the 25 *target* users as input. We test our CIDF algorithm using our collected dataset and compare it with Facebook’s FoF algorithm. Similar to our experiments on IDF-I and IDF-O, both CIDF and Facebook approaches are tested on the same set of users (10500).

A. VALIDATION METHODOLOGY

To accurately validate our algorithm, we will not use a list of FoFs of a given *target* user. This is because, in our algorithm, the FoFs do not have relationships with that user and, as a result, have no prior history of interactivity with that user. Instead, we use another approach to accurately validate the proposed algorithm. In this approach, we run the algorithm on each of the 25 users to recommend friendship connections from their already declared friends list who are also FoFs. For example, in Figure 5, user M is a friend of user x and also a *FoF* of user x because both x and M are friends of user J.

The algorithm has access to the relationship between user x and J and the relationship between M and J. The algorithm has no access to the direct relationship between x and M. Simply put, to the algorithm, user M is only an *FoF* of user x and we are using the actual friendship between x and M only to validate and measure the accuracy of our IDF algorithm.

B. EXPERIMENTAL RESULTS

In our experiments, we measure the accuracy of a friendship recommendation by its likelihood of becoming an

TABLE 6. Experimental results breakdown for algorithm CIDF.

Subgraph	# k Clusters	Available Friends	Total Recomm.	Interactive Recommended	%	Weak Eliminated	%
1	7	575	397	237	59.7%	137	46.13%
2	2	257	49	37	75.51%	161	93.06%
3	3	273	90	39	43.33%	154	75.12%
4	2	624	10	9	90.0%	560	99.82%
5	4	645	266	120	45.11%	295	66.89%
6	2	436	21	19	90.48%	328	99.39%
7	10	855	645	421	65.27%	155	40.9%
8	2	192	21	16	76.19%	124	96.12%
9	3	737	308	166	53.9%	352	71.26%
10	2	117	19	5	26.32%	83	85.57%
11	3	543	105	54	51.43%	395	88.57%
12	2	296	30	18	60.0%	239	95.22%
13	2	256	18	13	72.22%	214	97.72%
14	2	274	17	11	64.71%	218	97.32%
15	2	544	80	63	78.75%	298	94.6%
16	2	410	29	11	37.93%	342	95.0%
17	2	109	21	16	76.19%	55	91.67%
18	2	112	11	3	27.27%	92	92.0%
19	2	206	20	13	65.0%	162	95.86%
20	3	111	49	33	67.35%	40	71.43%
21	2	387	10	8	80.0%	327	99.39%
22	2	519	42	27	64.29%	435	96.67%
23	2	517	64	61	95.31%	380	99.22%
24	3	1027	147	67	45.58%	781	90.71%
25	2	453	28	8	28.57%	409	95.34%
Total		**10475**	**2497**	**1475**	**59.07%**	**6736**	**86.83%**

interactive relationship. Overall, out of the 2497 connections recommended by the CIDF algorithm, 1475 of them were of interactive relationships. This means that the accuracy of the CIDF algorithm to recommend interactive connections is:

$$\theta(CIDF) = \frac{1475}{2497} = 0.59 \tag{3}$$

As we can see in Table 5, the accuracy of CIDF is higher than the accuracy of IDF-I (0.57) algorithm, IDF-O (0.51) and Facebook's FoF (0.26). The percentage of recommending *weak* relationships (1-θ) is reduced from 0.74 to 0.41 which is better than both of the IDF algorithms.

As shown in Figure 6, Out of the 2791 interactive relationships, CIDF recommended 1475 interactive connections which is 53% of the total percentage of available interactive connections. Out of the 7758 *weak* connections within the dataset, the algorithm only recommended about 13% (1022 *weak* relationships) in comparison to 14% and 18% recommended by IDF-I and IDF-O algorithms respectively.

Each case presented in Table 6, CIDF algorithm recommended a higher percentage of *interactive* friends than Facebook. The CIDF algorithm was able to find considerably more *interactive* connections in cases 25, 18 and 10 than IDF algorithms. As we explained in Section VI, this is because our CIDF algorithm is independent of the number

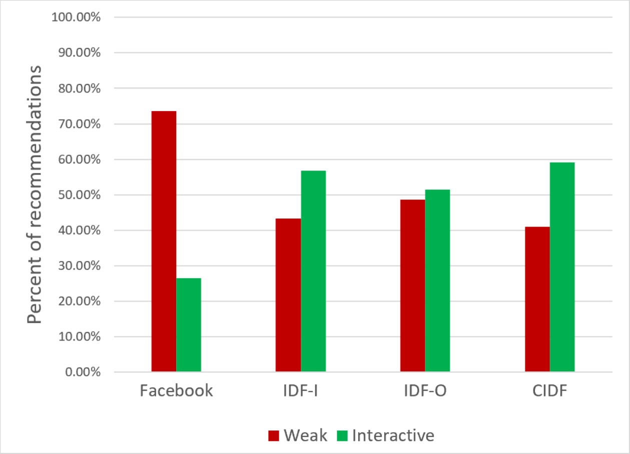


FIGURE 8. Summary of experiments.

of *commonFriends* and the *target* users in these three cases have a limited number of available *commonFriends*. The number of k clusters depends on the input. About 70% of our experiments on the 25 subgraphs required only 2 clusters and 20% required 3 clusters. Detailed results of the experiments are shown in Table 6. Figure 7 shows examples of clusters plots from the CIDF experiments. In most cases, the lowest mean cluster in each of the 25 experiments contains more users than the number of all users within the high mean clusters. This is not visible in Figure 7 below because most of the users in the lowest mean cluster are concentrated very close to the centroid.

VIII. CONCLUSION

Recommending a higher number of *interactive* connections can impact the overall accuracy of our IDF algorithms. As we have seen from our experiments on the IDF approaches that the IDF-O algorithm found 61 more interactive connections than the IDF-I algorithm. To find those connections, IDF-O recommended 316 more *weak* connections than IDF-I which affected IDF-O's accuracy. Our CIDF approach is able to overcome such a limitation by taking advantage of both of our IDF algorithms using k-means clustering algorithm. Figure 8 represent a comparison summary about the overall results of the Faccebook's FoF algorithm and our three algorithms IDF-I, IDF-O and CIDF.

In this paper, we identified and proposed a solution to the problem of lack of interactivity amongst connected users in online social networks. We also showed that the problem is caused by the fact that existing friending algorithms focus solely on generating easily accepted friendship connections. We developed an algorithm that generates easily accepted connections, but with a higher probability of leading to interactions. Our CIDF algorithm was able to recommend more than double the interactive friendships generated by Facebook's FoF algorithm. About 87% of the *weak* connections recommended by Facebook's FoF algorithm were also detected by our CIDF approach. By lowering the number of *weak* connections and increasing the overall percentage of interactive connections, more posts from interactive friends can be noticed. This leads to more interactions in online social networks. Our CIDF algorithm is built with the intention to offer meaningful relationships to users. These are relationships with a higher probability of exchanging communications and interactions which, in essence, is the ultimate purpose of a meaningful friendship.

REFERENCES

[1] R. E. Susskind and D. Susskind, *The Future of the Professions: How Technology Will Transform the Work of Human Experts*. London, U.K.: Oxford Univ. Press, 2015.

[2] J. Gantz and D. Reinsel, "The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east," *IDC iView: IDC Analyze Future*, vol. 2007, pp. 1–16, Dec. 2012.

[3] C. Desrosiers and G. Karypis, "A comprehensive survey of neighborhood-based recommendation methods," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2011, pp. 107–144.

[4] J. K. Kim, H. K. Kim, and Y. H. Cho, "A user-oriented contents recommendation system in peer-to-peer architecture," *Expert Syst. Appl.*, vol. 34, no. 1, pp. 300–312, 2008.

[5] Z. Zaier, R. Godin, and L. Faucher, "Recommendation quality evolution based on neighborhood size," in *Proc. 3rd Int. Conf. Automat. Prod. Cross Media Content Multi-Channel Distrib. (AXMEDIS)*, Nov. 2007, pp. 33–36.

[6] R. Burke, "Hybrid recommender systems: Survey and experiments," *User Model. User-Adapted Interact.*, vol. 12, no. 4, pp. 331–370, Nov. 2002.

[7] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.

[8] C. Battistella and F. Nonino, "Open innovation Web-based platforms: The impact of different forms of motivation on collaboration," *Innovation*, vol. 14, no. 4, pp. 557–575, 2012.

[9] Facebook. (2019). *Company Info*. Accessed: Aug. 21, 2019. [Online]. Available: https://newsroom.fb.com/company-info/

[10] *Number of Monthly Active Facebook Users Worldwide as of 2nd Quarter 2019 (in Millions) | Statistic*, Statista, Hamburg, Germany, 2019.

[11] C. M. K. Cheung, P.-Y. Chiu, and M. K. O. Lee, "Online social networks: Why do students use Facebook?" *Comput. Hum. Behav.*, vol. 27, no. 4, pp. 1337–1343, 2011.

[12] E. Courtin and M. Knapp, "Social isolation, loneliness and health in old age: A scoping review," *Health Social Care Community*, vol. 25, no. 3, pp. 799–812, 2017.

[13] J. Vines, G. Pritchard, P. Wright, P. Olivier, and K. Brittain, "An age-old problem: Examining the discourses of ageing in HCI and strategies for future research," *ACM Trans. Comput.-Hum. Interact.*, vol. 22, no. 1, 2015, Art. no. 2.

[14] E. K. Clemons, "The complex problem of monetizing virtual electronic social networks," *Decis. Support Syst.*, vol. 48, no. 1, pp. 46–56, 2009.

[15] A. Thomas and A. K. Sujatha, "Comparative study of recommender systems," in *Proc. Int. Conf. Circuit, Power Comput. Technol. (ICCPCT)*, Mar. 2016, pp. 1–6.

[16] R. Zhang, Q.-D. Liu, C. Gui, J.-X. Wei, and H. Ma, "Collaborative filtering for recommender systems," in *Proc. 2nd Int. Conf. Adv. Cloud Big Data (CBD)*, Nov. 2014, pp. 301–308.

[17] Y. Koren and R. Bell, "Advances in collaborative filtering," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2015, pp. 77–118.

[18] J. Ben Schafer, J. A. Konstan, and J. Riedl, "E-commerce recommendation applications," *Data Mining Knowl. Discovery*, vol. 5, no. 1, pp. 115–153, Jan. 2001.

[19] J. Chen, W. Geyer, C. Dugan, M. Muller, and I. Guy, "Make new friends, but keep the old: Recommending people on social networking sites," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, 2009, pp. 201–210.

[20] P. Lops, M. de Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2011, pp. 73–105.

[21] S. A. Golder, D. M. Wilkinson, and B. A. Huberman, "Rhythms of social interaction: Messaging within a massive online network," in *Communities and Technologies*. London, U.K.: Springer, 2007, pp. 41–66.

[22] C. Wilson, B. Boe, A. Sala, K. P. N. Puttaswamy, and B. Y. Zhao, "User interactions in social networks and their implications," in *Proc. 4th ACM Eur. Conf. Comput. Syst.*, 2009, pp. 205–218.

[23] L. M. Verbrugge, "The structure of adult friendship choices," *Social Forces*, vol. 56, no. 2, pp. 576–597, 1977.

[24] H. Alves, A. Koch, and C. Unkelbach, "My friends are all alike—The relation between liking and perceived similarity in person perception," *J. Exp. Social Psychol.*, vol. 62, pp. 103–117, Jan. 2016.

[25] R. E. Wilson, S. D. Gosling, and L. T. Graham, "A review of Facebook research in the social sciences," *Perspect. Psychol. Sci.*, vol. 7, no. 3, pp. 203–220, May 2012.

[26] A. Alshammari and A. Rezgui, "Better edges not bigger graphs: An interaction-driven friending algorithm for the next-generation social networks," Dept. Comput. Sci. Eng., NMT, Socorro, NM, USA, Tech. Rep., 2019.

AADIL ALSHAMMARI received the B.S. degree in computer science from Qassim University, KSA, in 2006, and the M.S. degree in information systems from the University of Surrey, U.K., in 2011. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, New Mexico Institute of Mining and Technology, USA. From 2011 to 2016, he was a Lecturer with the Department of Computing and Information Technology, Northern Border University. His research interests include recommender algorithms, social networks, clustering algorithms, and data analytics.

ABDELMOUNAAM REZGUI received the Ph.D. degree in computer science from Virginia Tech. He is a Faculty Member of the School of Information Technology, Illinois State University. He has authored or coauthored over 80 articles in top journals and conferences including the IEEE TBD, the IEEE TKDE, ACM TOIT, the IEEE TPDS, the IEEE Internet Computing, the IEEE Security and Privacy, the IEEE ICDE, the IEEE IC2E, and the IEEE Cloud Computing. His research interests include graph algorithms, networking, cloud computing, big data, and data analytics. His research has been funded by the NASA, the NSF, the ICASA, and Microsoft. He regularly serves on the program committees of several major conferences including the IEEE Big Data, the IEEE BigDataSE, the IEEE GLOBECOM, the IEEE CloudNet, the IEEE LCN, and IoTBDS. He has also been a proposal Reviewer and a Panelist of NSF and QNRF. He has been the Track Chair, a Keynote Speaker, or a Tutorial Presenter at several international conferences. He is also on the Editorial Board of several journals including *Big Data Analytics* (Springer). His research conducted by his students has been invited for presentation at a very selective events organized by major organizations including NSF, Google, and Siemens.