Discovering social circles in a directed social network using node, structure and edge features

Joyce Yang
jingyi-yang@uiowa.edu
University of Iowa
Iowa City, Iowa, USA

Muneeb Shahid
mushahid@uiowa.edu
University of Iowa
Iowa City, Iowa, USA

ABSTRACT

Online Social Network (OSN) platforms enable users to organize their social circles for managing their contacts. Organizing social circles is a manual task which is not only tedious but unscalable and in-effective as well. Automation of this process would be beneficial for the users to fulfill their social and professional requirements. Various approaches have been proposed for the automation of this task but none of these approaches takes edge features like trust level between the nodes alongside with other node and structural features into account. In this project, we propose the Node- Edge K-means clustering algorithm to study the importance of individuals with a good reputation in building social circles. Our approach builds on top of the Genetic Algorithm variant of K-means clustering algorithm that considers only node and structural features for discovering social circles. With the help of intrinsic measures, we then evaluate our approach by comparing the quality of social circles discovered after adding the trust level feature with those discovered without this feature.

ACM Reference Format:
Joyce Yang and Muneeb Shahid. 2020. Discovering social circles in a directed social network using node, structure and edge features. In *Iowa City '20: Mining and Learning on Large Networks, October 15–16, 2020, Iowa City, IA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Online Social Networking platforms like Facebook, Twitter, and etc generates a large amount of content as these platforms are becoming popular. Users of these platforms are interested in sharing the information and content with their peers and friends. However, different groups of people have different preferences for the content they want to receive. From a user's perspective, privacy and security is another reason for selective content sharing. To address this issue of selective content sharing, most of the OSN platforms gives the user the ability to organize their contacts into different groups (Social Circles). However, this feature is not popular among the users. The manual nature of this task makes it tedious and ineffective for the user to perform it. As time passes the user's network grows and it becomes in-feasible for the user to keep social circles up to date hence, at any given time the user's social circles in OSN do not represent the ground truth. Information overload managing, friends recommendation and prediction, achieving professional and social objectives are few reasons other than selective content sharing that highlights the importance of organizing and managing social circles. Efficacy of this task can be increased by a process called social circles discovery which automates the task of managing social circles. Social Circles discovery problem can be seen in tandem with the problem of clustering of nodes in Ego Social Network (ESN). In ESN, the ego is a user and its connection with the network of other users (friends) is defined as alters.

A variety of approaches have been proposed to discover social circles throughout recent years. All of these approaches use a certain set of features for social circles discovery. Some of them prioritize node-level features while some of them consider topological features to be most important. These features can be considered individually or can be aggregated by assigning some degree of importance to each variable. Few of such approaches are discussed in [1], [4],[6], and [10]. However, each feature affects the social discovery process differently. This gives us the motivation to study the importance of each feature in the social discovery process in detail.

Studying importance of individual's behavior within the given network in discovering social circles is a new research direction. One of the features that helps in determining individual's behavior is trust. In this project, we propose an approach called Node-Edge K-Means Clustering Algorithm that incorporates trust between connected nodes / individuals in the process of social circle discovery. With the help of this approach, we will be able to study the importance of stature of individuals in forming social circles and we might be able to discover small circles that can not be discovered without considering the trust level between connected nodes. Our approach is based on the K-means clustering algorithm and uses the genetic algorithm for identifying important centroids which are active centers or nodes of creation in the context of this project. Our approach is built on top of a Genetic Algorithm variant of the K-means clustering algorithm [1] that takes only node level and structural features into account.

2 PROBLEM DEFINITION

Since, our motivation is to find social circles in user's social network we assume that we are given a signed directed graph comprising of friends of any user of the OSN platform.

Let us consider a directed social network G(V,E,A), where V is the set of nodes, E represents the set of edges and A is the attributes vector for every node in set V. Each edge has a positive or negative sign. Positive signs show trust, while distrust is indicated by

Iowa City '20, October 15–16, 2020, Iowa City, IA

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

negative sign. Every node in the network contributes differently towards the development of the social circle and the amount of contribution depends on their characteristics. In this project, we have to find such nodes that strongly hold other members in the circle together, shares large number of common attributes with other nodes in the circle and has a most trustworthy reputation in the network. As mentioned earlier these nodes will act as active centers. The quality of the social circles formed depends on the selected set of active centers. For this purpose, we are going to calculate the values of measures related to node's attributes, relationships, level of interactions and trustworthiness. For example, if we randomly choose any one node N from the network then we have to compute measures for N using remaining nodes in the network. Then based on these computed measures we have to identify nodes that are good candidates to become the member of the social circle for that node N and filter those nodes from other nodes for node N. For any node N, our objective is to minimize the difference of measures between the nodes within its circle and maximize the difference for nodes outside its circle. Based on this objective we need to select best predefined number of active centers for given directed network. After discovering the social circles we have to evaluate the circles using standard evaluation metrics and compare their quality with the circles formed without incorporating trust level feature.

This problem should not be confused with the traditional community detection problem. Traditional community detection algorithms are not well-suited for social networks. Structure of social networks is sparse therefore, it is difficult to establish global knowledge of the network. In community detection, the focus is on the identification of highly interconnected subgraphs within a large network while our problem focuses on finding a single subgraph in a network based on the selected set of nodes. The role of these nodes is to provide some level of context to the search. Importantly, most data mining algorithms for community are developed for undirected graphs and it is assumed that they can be applied to directed graphs as well by ignoring direction but, this results in loss of valuable information especially in case of social networks. Also, incoming links to node may not be known without performing exhaustive search in graph which makes the approach of these traditional algorithms inadequate. However, our problem deals with directed social networks specifically by considering the importance of direction of edges between nodes.

3 RELATED WORK

Different approaches take advantage of different types of features in the network for discovering social circles. [10] considers structural and node features together and it proposes an algorithm called FESC. FESC is based on a unified mathematical model that takes structural information of the network alongside attributes information of the alters into account. FESC does not require global information of the network as it proceeds with the assumption that homophily exists in the network and local information is sufficient for it to work accurately.

The algorithm proposed in [4] finds social circles by taking edge directions into account, unlike most existing community algorithms. It also avoids including high profile nodes like celebrities and unknown nodes that do not have mutual interaction with the social circle. Through this approach, social circles will be discovered based on ground realities.

The data collected from OSN platforms does not contain underlying strength of relationships between the users. [3] performs the task of social circles discovery by measuring strength of relationship between the nodes. For this purpose, it considers five classes of variables 1) Text-based communication. 2) Like-based communication. 3) Homophily. 4) Time / Duration. 5) Structural dimensions.

Node attributes and the network topology are two different types of sources of information that can greatly affect the performance of clustering algorithms. Using the information at the node level is helpful in clustering nodes with similar attributes, but the nodes with missing profile information will not be correctly clustered. On the other hand, the information extracted at the network level will be useful to understand the presence of different relationships among the nodes in the network but the nodes with fewer connections will not be classified correctly. [6] proposes two separate approaches using Ant Colony Optimization (ACO) based clustering algorithm to address these problems. ACO algorithm is based on the foraging behavior of the ants. Both versions of ACO algorithms, topology-based and attributes based, are determined using a reduced decision graph and their heuristic functions are also different.

All of these traditional approaches focus on common characteristics of nodes within the circle. In contrast to traditional approaches, [1] discovers circles by finding nodes with high social affinity and then builds social circles around these nodes. It calculates scores for nodes concerning profile and structural attributes. However, none of the above-mentioned approaches considered an individual's behaviors in determining social circles. We believe that individual behaviors such as trust among users could have a strong contribution to the determination of social circle. [2] discusses the role of trust in information diffusion which potentially gives us the motivation to go forward with our idea.

4 PROPOSED FRAMEWORK

GA K-Means Clustering model as mentioned in previous section is a genetic algorithm that predicting the dynamics of social circles in undirected ego networks using pattern analysis, such as network structure and node features. And this leads us to think about the impact of trust information in edge and the social circle predictions for directed ego networks. Besides the intuitions, like degree centrality, mentioned in GA K-means clustering methods, we think that the positive or negative directed link should also affect the result. Therefore, we come up with Node-Edge K-means clustering methods to predict social circles in directed networks that involves analyzing the network pattern, the node features and link features. Following sections will walk you through each part of Node-Edge K-Means algorithm and explains the intuition of that part.

4.1 Node-Edge K-Means Clustering

The GA K-means clustering algorithm uses the embeddedness of nodes to represent the level of interactions a node has with the members of a group and with those existing outside the group. It analyzes these features by calculating degree centrality, profile

similarity, and strength of ties. However, this algorithm only fits with an undirected graph. To apply the algorithm on a directed network graph, we first modify the GA K-means clustering algorithm without considering the link features. A new definition in the context of directed graphs for degree centrality, profile similarity, strength of ties, trust info, residual and core area is defined in the following content.

4.1.1 Degree Centrality. It is defined as the number of connections a node has in an undirected network, which is the number of links incident upon the node. However, for a directed graph, we define it as the number of connections that node has in-network, which is the number of both in and out links incident upon the node. The formula of degree centrality (deg_cen) of node x in circle c is defined as:

$$deg\_cen_x^c = \frac{(\#\ of\ Link_{in}^c(x)\ +\#\ of\ Link_{out}^c(x))}{(n-1)} \tag{1}$$

where n is the number of members in circle c, $Link_{in}^c(x)$ means links that point into x from members of circle c and $Link_{out}^c(x)$ means links that point into x from members of circle c.

Since degree centrality is analyzing the degree of condensed network, our intuition is that a good social circle should have a strong interaction between users other than a sparse connection. Therefore, the larger the $deg\_cen_x^c$ is, the connection among all members in the circle is stronger.

4.1.2 Profile Similarity ($prof\_sim$). It counts the number of attributes a node has in common with others. Because this is not affected by directed links, the definition of it is almost the same with slight modification. Profile similarity of a node x with a node j is computed as:

$$prof\_sim(x,j) = \frac{\sum_{i=1}^{p}\delta_i(x,j)}{(p)}, \tag{2}$$

where p is the number of attributes of a node. However, the definition of $\delta_i(x,j)$ is different from the GA K-means clustering algorithm because the dataset used in [1] has categorical features and it determines profile similarity based on the number of common feature values between two nodes. While, in our case, the node2vec algorithm creates quantitative features for the Slashdot Social Network dataset. Therefore, instead of making $\delta_i(x,j) = 1$ when the attribute is presented in the profiles of both x and j, we determine profile similarity based on the sum of squared distance between feature values of nodes. The formula is defined as:

$$\delta_i(x,j) = \frac{1}{\sqrt{\sum_{i=1}^{p}(x_i - j_i)^2}}, \tag{3}$$

where x_i represents the ith feature value for node x and p is the number of features in each node. The more similar two nodes are, the smaller value of the denominator will be in equation (3). And by taking the inverse of it, nodes with higher similarities will have a higher value in $prof\_sim$.

Our intuition of using profile similarity is that people with the same node features, such as similar hobby, similar location tend to be one social group. Therefore, the higher $prof\_sim(x,j)$ tends to provide a better social clusters.

4.1.3 Strength of Ties (str). It computes the degree of closeness of the relationship between two nodes. It is borrowed from the idea of base node similarity, where the strength of ties between two concerned nodes is determining by calculating the number of links each node has with other neighboring nodes, a large number of such links indicates less value for the strength of ties between them. Because of the directed graph, the degree of node x is then divided into $degree_{out}$ and $degree_{in}$ of the node. Therefore, we reformat the equation into:

$$str(x,u) = \frac{1}{deg_{out}(x) + deg_{in}(u) - 1} \tag{4}$$

where $str(x,u)$ stands for the strength of the link that node x points to node u.

Our intuition of considering strength of ties is that people can only have certain amount of attentions, therefore, the more attentions you give, the less each one will receive. Similarly, the more attention you receive, the less you will care about for each attention. For example, if a person only have one friend, then the connection between the person and his friend is strong. In other word, the weight of this ties should be viewed more precise. However, if a person has many friends, then he may not know one well enough. Similarly, a famous star may not care the opinion online as much as an individual because the famous star has a lot of attentions from his fans, therefore, one bad opinion about him will not hurt him that much. However, if a simple person receive a critic online, he may care so much because this is one of the few judgements he received.

4.1.4 Link Info. The link info represents one's opinion towards the other. The link info can either be negative or positive. A negative link from node u to node x means u has a negative point of view towards x. And a positive link from node u to node x means u has a positive point of view towards x. The negative point of view can be described as distrust, not interested in the content and etc. Similarly, positive point of view can be described as trust, interested in content and etc. Normally the value of link info should be either 1 or -1 but a decimal is also accepted.

Our intuition about apply link info is like what we mentioned in previous section. The attitude between two users should also be considered when forming a social circle. For example, people may have similar features, such as same hobby or same location but that do not mean that they are friends. People with high similarity tend to dislike each other because they act more like a substitution of each other. For example, two same age students interested in algorithms may attend ACM competition together. However, only one can win the competition. In this case, they may not want to share their information with each other, not even to consider them as friends. Therefore, a positive view between two nodes should up-rise the social circle, while a negative view between two nodes will not.

4.1.5 Residual and Core Area. The entire network is divided into two parts. One is *Core Area(C)* and the other one is *Residual Area(R)*.

Core Area: is the member of the circle.

Residual Area: is the immediate neighbor of any of the member of the circle and provide interaction information with members of the circle.

The intuition of dividing the network into two parts is trying to observe some characteristics in the network. In order to have a good social circle clusters, we want:

- Members in the circle is strongly combined with each other, while it is loosely coupled with members outside the circle.
- Members in the circle should have a higher profile similarity value with all members of the circle than members outside circle.
- With regard to the strength of ties, ties between the core area should be strong and ties between the residual area should be relatively weak.

4.2 Algorithm

Since all the concepts used in the algorithm have been defined in the previous section, in this section, We will explain our algorithm for Node-Edge K-means clustering.

4.2.1 K-means Clustering. First of all, we design the algorithm (Algorithm 1) to allocate nodes in a circle concerning their affinity with certain active centers based on K-means clustering. We have one active center for each circle and the circles are discovered for these active centers. Algorithm 1 is not the same as the GA K-means clustering algorithm because we are considering the directed graph and we incorporate the trust feature into this algorithm as well. p_1 represents whether the active center points to the node, p_2 represents whether the node points to the active center, p_3 represents profile similarity between node and active center, p_{4_1} represents the strength of ties between active center and node,p_{4_2} represents the strength of ties between node and active center, p_{5_1} is the trust level that active center have to node, p_{5_2} is the trust level that node have to active center. max_s is a new measure standard that takes profile similarities and strength of tie equally into account.max_s is the sum of all previous value p_3, p_{4_1}, p_{4_2}, p_{5_1},p_{5_2}. The node will be assigned to the center that generates a maximum value of max_s. With the variable of p_1, p_2, p_{4_1},p_{4_2},p_{5_1},p_{5_2}, we separately considered the direction issue. This will performs better than the origin because we track the profile similarities, strength of ties and trust information into account. To maximize the max_s, we will be able to have clusters that has strong connection, high similarities and high agreement on each other. We understand that the formulation is not complex. There are two other ideas that we also think about:

- $max_s \leq p_3 + p_{4_1} * p_{5_1} + p_{4_2} * p_{5_2}$. With this function, we consider the attitude towards each other as an important fact. In this case, any bad attitude will have a heavy impact on the result. However, we do not want this. We want to make sure that the attitude between two nodes has some effect but not determine the outcome. For example, if p_{5_1} is negative then the whole $p_{4_1} * p_{5_1}$ will decrease the criteria value. However, with the addition, even though we have a bad attitude towards other, a higher value on other aspect may off-set the badness of attitude.
- $max_s \leq \alpha_1 p_3 + \alpha_2 p_{4_1} + \alpha_3 p_{5_1} + \alpha_4 p_{4_2} + \alpha_5 p_{5_2}$ This may actually be a better idea however to get the variable, our algorithm will turn into a supervised algorithm. Currently, we are missing the target label information in our dataset. Therefore, we can not try this equation.

Algorithm 1: Assign social circles for each node

Input: A chromosome consists of k active centers $X = x_1, x_2, ...x_k$, and set of users $U = (V - X)$ to be clustered.
Output: set of predicted circles $\{\hat{C_1}, \hat{C_2}, ..., \hat{C_k}\}$
while $U \neq \emptyset$ **do**
  pick up a node $u \in U$ **initialize:** $max_s = 0,\ AC = -1$
  for $i = 1$ *to* k **do**
    $x_i \in X$
    initialize: $p_1 = 0,\ p_2 = 0,\ p_3 = 0,\ p_{4_1} = 0,\ p_{4_2} = 0,\ p_{5_1} = 0,\ p_{5_2} = 0$
    Compute values of p_1, p_2, p_3, p_{4_1},p_{4_2},p_{5_1},p_{5_2} between u and the active centre x_i
    if $max_s \leq p_3 + p_{4_1} + p_{4_2} + p_{5_1} + p_{5_2}$ **then**
      $max_s = p_3 + p_{4_1} + p_{4_2} + p_{5_1} + p_{5_2}$
      AC = x_i
    end
  end
  if $AC \neq -1$ **then**
    Add node u in the cluster of AC
  end
end

4.2.2 Objective Function. Now we have the function to discover social circles around each active center; however, the quality of social circles depends on the onset of active centers we select. Therefore, we decide to use fitness value as an evaluation of quality. The fitness value is calculated based on the objective function and the objective function calculates values for degree centrality, profile similarity, the strength of ties, and newly added trust features concerning member inside the circle and the members outside the circle and Objective function is defined as follows:

$$Obj(j) = [deg\_cen_j^C(x_i) - deg\_cen_j^R(x_i) + prof\_sim_j^C(x_i) - prof\_sim_j^R(x_i) + str_j^C(x_i) - str_j^R(x_i) + trust_j^C(x_i) - trust_j^R(x_i)] \quad (5)$$

$$fitness = f(X_i) = \frac{\sum_{j=1}^{k} Obj(j)}{k} \quad (6)$$

where R indicates "Residual Group". K is the number of centers in one set, similar to the idea introduced in the beginning of this section:

- $deg\_cen_k^C(x)$ and $deg\_cen_k^R(x)$ means the average connectivity coefficient of x with the members of $C^{[k]}$ and $R^{[k]}$ respectively.
- $prof\_sim_k^C(x)$ and $prof\_sim_k^R(x)$ means the average profile similarity of x with the members of $C^{[k]}$ and $R^{[k]}$ respectively.
- $str_k^C(x)$ and $str_k^R(x)$ means the average strength of ties of x with the members of $C^{[k]}$ and $R^{[k]}$ respectively.
- $trust_k^C(x)$ and $trust_k^R(x)$ means the average trust level between x and members of $C^{[k]}$ and $R^{[k]}$ respectively.

4.2.3 Genetic Algorithm. Genetic algorithm (GA) [1] can be served as the optimizing tool for the initial seeds in K-means clustering. Genetic algorithms are stochastic search techniques that can search

for large and complicated spaces. For example, crossover (Figure 1) and mutation (Figure 2) are common methods and our Node-Edge K-means clustering algorithm implement these two methods as well.

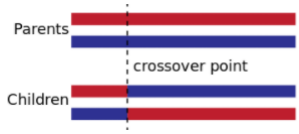

Figure 1: Example of Crossover

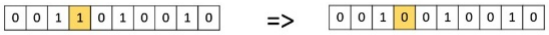

Figure 2: Example of Mutation

After calculate the fitness value for the generated N set of active centers where each set has k active centers, We use the genetic algorithm approach to avoid randomly selection. We will apply crossover and mutation operations to generate the best set of active centers and the quality of active centers is determined by fitness value. Individuals with high fitness value are more likely to be selected as parents and generate offspring by means of crossover and mutation. Therefore we pick two sets with the higher fitness value, we then can generate offsprings from these two set through crossover and mutation. Sets with a better fitness value are promoted to increase the probability of their selection for the next iterations. We continue to repeat this process until convergence of fitness of value. In our approach, we check for consecutive 10 iterations without any change in the best fitness value for convergence (Algorithm 3).

Algorithm 2: Calculate fitness value for a set of k active centers

Input: Two N x k matrices $[x_{ij}]_{Nxk}$ and $[\hat{C_{ij}}]_{Nxk}$, N: number of population, k: number of circles

Output: Best possible set of k seeds x_1,x_2,...,x_k for Ego network segmentation

for 1 *to N* **do**

  initialize: $fitness = 0$

  pick up the i^{th} row from $[x_{ij}]_{Nxk}$ and $[\hat{C_{ij}}]_{Nxk}$ **for** 1 *to k* **do**

    initialize: $obj = 0$

    $Obj(j) = [deg\_cen_j^C(x_i) - deg\_cen_j^R(x_i) + prof\_sim_j^C(x_i) - prof\_sim_j^R(x_i) + str_j^C(x_i) - str_j^R(x_i) + trust_j^C(x_i) - trust_j^R(x_i)]$

  end

  $fitness = f(X_i) = \frac{\sum_{j=1}^{k} Obj(j)}{k}$

end

Algorithm 3: Find the best set of active centers for social circles discovery

Consider population matrix $pop(X) = [X_i]_{Nx1} = [x_{ij}]_{Nxk}$ and fitness matrix $F = [f(X_i)]_{1xN}$ and generate an augmented matrix $Q = [X_i|f(X_i)]_{Nx(k+1)}$

Then sort matrix Q in descending order of fitness value $f(X_i)$

while *changes in highest fitness value during* 10 *consecutive iterations appears* **do**

  for 1 *to N* **do**

    1. Randomly select two parent chromosomes with relative high fitness value. For example, from top 10 of the descending Q.

    2.**[CROSSOVER]:**

    Generate a random (integer) number *randc_pos* from the range [1,k], and exchange the alleles of chromosomes X_1 and X_2 at random position(*randc_pos*) to produce two new chromosomes X_1^{new} and X_2^{new}

    compare the fitness of X_1, X_2, X_1^{new}, X_2^{new} and feed the one with best fitness value to mutation

    3.**[MUTATION]:**

    Generate a random position *randm_pos* in the range [1,k] and *rand_id* in the range [1,n], then mutate the allele which is at *randm_pos* by *rand_id*.

    Compare the X^{new}'s fitness value with the Q_i 's fitness value.

    $Q_i \Leftarrow X^{new}$ if X^{new} has a better fitness value

  end

  Sort matrix Q in descending order of fitness value $f(X_i)$ again for next round.

end

4.2.4 Summary. The main steps of Node-Edge K-Means Clustering is presented as under:

(1) Input the network information
(2) Generate initial random population size N and the number of active center k.
(3) Input these sets of active centers into algorithm 1 to receive the clusters for each active centers.
(4) Calculate the fitness value for each sets of clusters calculated based on active centers through algorithm 2.
(5) Sort the population based on the fitness value and apply mutation and crossover as described in Algorithm 3
(6) Repeat Step 3 to Step 5 until the highest fitness value remains the same in 10 iterations

4.2.5 Optimization. We code this algorithm in python. Recall that our algorithm repeat Step 3 to Step 5 (mutation and crossover among high fitness value) in order to optimize the objective function, it takes time to converge. Therefore, we apply parallel computing with python package multiprocessing to Algorithm 3. Since each

set of active centers is independent with other set, we can calculate the fitness number for each set of active centers separately and then wait until all set finished. Then apply the crossover and mutation again and parallel the next iteration.

5 EXPERIMENTS

We performed some experiments to evaluate the social circles formed by our approach and compare them with the social circles formed without incorporating trust level feature. We did experiments using signed network dataset.

5.1 Data

The ideal dataset should include multiple node features, link features like trust information, and a clustering label. However, we didn't find public datasets that satisfy all the requirements. Therefore, we decided to go with the Slashdot Social Network dataset [8] and apply further processing methods on it to make the Slashdot Social Network dataset suitable for our requirements.

Slashdot is a website for social news where news stories related to politics, technology, and other sectors of life are submitted. These stories are then evaluated by users and editors. The Slashdot Social Network dataset consists of a directed graph where each edge is signed. The positive sign indicates the trust between the two nodes and the negative sign represents distrust between the two nodes. This dataset does not contain node features and to generate node features we have run node2vec algorithm [7] on this dataset. Node2vec is an algorithm to generate low dimensional features representation for each node in the network with regards to the topology of neighborhoods. With the help of this algorithm, we have generated 100 features for each node. After running node2vec, we have a dataset consisting of the node, structure and edge features.

Table 1: Dataset Statistics

Property	Value
No. of Nodes	82140
No. of Edges	549202
Avg. Clustering co-efficient	0.0588
Diameter	12
No. of features for each node	100

5.2 Parameters

The performance of our approach relies heavily on the selection of active centers or centroids in terms of K-means clustering. Before selecting a good set of active centers we need to know what is the optimal number of active centers or value of K for a given dataset. Therefore, we have used the elbow method to run the K-means clustering algorithm on the dataset for the given range of values of K from 1 to 50. For each value of K, it calculates the sum of squared differences and plots the data points accordingly. The resulting plot is shown in Figure 1. Using this plot we identified that 7 would be the appropriate value of K, in other words, there are 7 suitable amounts of clusters in the dataset. We are going to use this value for our experiments as well.

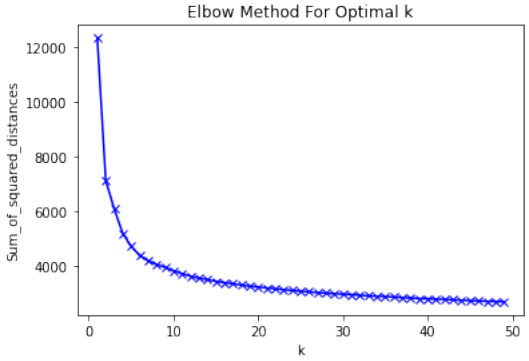

Figure 3: Optimal number of social circles / active centers

Next, we pick a reasonable amount of population size N=20 for algorithm 2 and 3.Finally, to determine a relatively high value of fitness value, we randomly picked 2 sets of active centers in top 10 of the populations and apply mutation and crossover on it.

5.3 Evaluation Metrics

As mentioned previously, our dataset does not have any ground-truth labels associated with it. Without labeled data, standard measures like prediction accuracy, balance error rate, precision, recall, and the f-score can not be calculated. We decided to compare the social circles formed by calculating net values of the properties (Degree Centrality, Strength of ties, profile attributes). We calculated these values for all the members or nodes with respect to each of the selected active center. Nodes that were not included in the social circle of some active center are put into group called residual group, and this group like social circle will be specific for each of the selected active centers. We calculates the values of degree centrality, strength of ties and profile similarity for both the social circles and residual groups formed around each of the selected active center. Difference between the values of these properties for social circles and residual group will help us evaluate our approach. Higher the difference better is the quality of social circles formed.

Other than using properties, there are ways to evaluate goodness of social circles formed. These methods can be divided into two categories, one is external validation and other is internal validation. We cannot use external validation methods due to absence of labelled data. We used internal validation measures like Silhouette Coefficient, Calinski Harabasz, and Davies Bouldin to determine the goodness of the social circles formed by our approach [9]. Generally, the purpose of experiments was to answer following questions:

- Does the proposed method increase or decrease the difference between the net objective values of properties of social circles and residual groups?
- Does the proposed method yields higher Silhouette Coefficient index Score?
- Does the proposed method yields lower Davies Bouldin index Score?
- Does the proposed method yields higher Calinski Harabasz index Score?

5.3.1 Net objective values of properties. Following are the calculated net values for individual properties and overall differences in the values:

Table 2: Net values of the properties

Property	No Trust	Trust
Net Degree Centrality for Social Circle	1.35	1.20
Net Degree Centrality for Residual	0.22	1.17
Net Strength of Ties for Social Circle	0.20	0.05
Net Strength of Ties for Residual	0.0008	0.02
Net Profile Similarity for Social Circle	0.06	0.06
Net Profile Similarity for Residual	0.009	0.05
Net Objective Value (overall difference)	1.38	0.07

From above results, it can be observed that difference between the net values for social circles and residuals have decreased for every property resulting in decreased net objective value. Incorporating trust feature does not yield better social circles especially in terms of these properties. It looks like trust level feature is drawing those nodes away from the social circles which could be a good candidate to become a member of the circle.

5.3.2 Silhouette Coefficient index. This index helps us in evaluating performance of our approach in determining social circles by taking pairwise difference of between and within-circle distances. The value for this index ranges from -1 to 1. Higher score indicates that circles are dense and well separated [11]. Following is the formula for calculating this index:

$$s = \frac{b - a}{max(a, b)} \tag{7}$$

Where a is the mean distance between a node and all other nodes in the same circle and b is the mean distance between a node and all other nodes in the next nearest cluster.

Table 3: Silhouette Coefficient index scores

Property	No Trust	Trust
Silhouette Coefficient Score	-0.20	-0.20

Unfortunately, these scores does not give us the conclusive evidence to determine whether trust feature increased or decreased the quality of social circles. Since, all of the nodes features are quantitative values so it should be no surprise that we yield same values for net profile similarity as well because in calculating net profile similarity we also computed the distance based on the features and similar approach is followed by Silhouette Coefficient index. If we had any categorical or nominal features the we would have calculate net profile similarity by boolean matching of such feature values.

5.3.3 Davies Bouldin index. This index computes the average similarity between social circles, where the similarity is a measure that compares the distance between circles with the size of the circles themselves. Lower the value for this index greater the difference between the circles in other words the resulting social circles are better [5]. Following are the formulas for calculating this index:

$$R_{ij} = \frac{s_i - s_j}{d_{ij}} \tag{8}$$

Where, s_i is the average distance between each point of circle i and the active center of that circle and d_{ij} is the distance between active centers of circle i and j.

Finally, Davies Bouldin index is defined by,

$$DB = \frac{1}{K} \sum_{i=1}^{K} max(R_{ij}) \tag{9}$$

Where K, is the total number of social circles.

Table 4: Davies Bouldin index scores

Property	No Trust	Trust
Davies Bouldin Score	6.80	6.03

This index shows results in favor of trust feature. We have lower Davies Bouldin index value with trust feature than without trust feature. This gives us some level of intuition that with trust feature we are able to generate more distinct social circles. This is important in terms of application of discovering social circles. Greater the difference between the circles means that there are less number of nodes that overlaps with different circles. This is useful for applications like selective content sharing where user will be able to filter content more confidently. User will be less worried about keeping some friends devoid of relevant information and sharing content with friend for whom the information is irrelevant.

5.3.4 Calinski Harabasz index. This index is the ratio of the sum of between-circles dispersion and of inter-circles dispersion for all circles. In our case the dispersion is defined as the sum of distances squared. Higher value for this index means better quality of social circles [**?**]. Following is the mathematical definition of this index (s):

$$s = \frac{tr(B_K)}{tr(W_K)} \frac{n_E - K}{k - 1} \tag{10}$$

Where, E is a set of data and n_E is the size of the data which has been divided into K number of circles. $tr(B_K)$ is trace of the between circle dispersion matrix and $tr(W_K)$ is the trace of the within-circle dispersion matrix, where B_K and W_K are defined by,

$$W_K = \sum_{q=1}^{K} \sum_{x \epsilon C_q} (x - c_q)(x - c_q)^T \tag{11}$$

$$B_K = \sum_{q=1}^{K} (n_q)(c_q - c_E)(c_q - c_E)^T \tag{12}$$

Where, C_q is the set of nodes in the circle q, c_q the center of cluster q, c_E the center of E, and n_q the number of nodes in the circle q.

Table 5: Calinski Harabasz index scores

Property	No Trust	Trust
Calinski Harabasz Score	36.6	57.02

This index also yields results in favour of trust level feature and there is a significant difference between the resultant values. Value with trust feature is far more greater than the value without trust feature. This validates our intuition that we developed in interpreting Davies Bouldin Scores, that trust level contributes towards the development of dense circles that are different from each other at least in terms of node level attributes.

6 CONCLUSION AND FUTURE WORK

Online Social Network (OSN) platforms allows users to organize their social circles. This manual task is not only tedious but ineffective as well. Also, like networking in real life, in OSN, users are eager to grow their list of friends due to various social or professional reasons. As the list of friends grows, it becomes arduous for the users to keep the list of groups up to date hence, these lists become stale and does not represent ground reality. However, importance of organizing and managing social circles cannot be sidelined as it has various applications like content filtering, managing information overload, friends recommendation and prediction, etc. Efficacy of social circle development can be increased by its automation. Various approaches have been proposed for the automation of this task but none of these approaches considers edge features like trust level between the nodes into account alongside with the other features. Our proposed approach incorporates trust level feature in Genetic Algorithm variant of K-means clustering algorithm to study the importance of individuals with good reputation in building social circles. Along with the trust level feature we have taken some node and structural attributes into account as well. These attributes include profile features, degree centrality and strength of ties.

After discovering social circles with our approach, we have evaluated the quality of social circles formed and compared the quality with social circles discovered without trust level feature. For this purpose, we have calculated net objective values with respect to features other than trust. In terms of features values we didn't see any improvement in the quality of social circles formed with trust feature instead, the quality decreases. Other than evaluating features, we used intrinsic measure for our experiment as well. Sillhouette Coefficient index, Davies Bouldin index and Calinski Harbrasz index are the intrinsic measure for measuring the goodness of social circles formed. Analysis of the values these indexes gave promising results in favour of trust level feature and we came to conclusion that trust plays an important role in discovering social circles that are dense and distinct from each other. However, we cannot make a definitive statement that trust level features definitely increases the quality of social circles. For some applications like selective content sharing we can say that it does yield well defined and well separated circles but for other type of applications especially those that gives more importance to structural attributes of the circles, incorporating trust level might not be a good option.

For the future, we can edit our algorithm by determining different weightages for the features based on the desired application. In this way our approach would be able to generate circles based on given application, as mentioned earlier, some applications gives more weightage to some features over others. Additionally, graph summarization of the social circles would be an interesting option, with this approach we would be able to define each social circles in terms of some value. This will be helpful in generating labels for underlying social circles in the network and these labels can be used for performing accuracy and error evaluation measures like precision, recall and etc, which we couldn't do due to absence of ground truth labels.

Besides, applying social behavior theory should also be a great directions. Social behavior theory, such as "my enemy's enemy is my friend","my friend's enemy is my enemy" etc.,not only provides a good link prediction direction but also predict a great potential social circles. By taking the triad relationship and information in to the node features or link features, our social circles should be better.

7 INDIVIDUAL CONTRIBUTIONS

7.1 Jingyi Yang

With the great project idea from Muneeb, I designed the algorithm based on GA K-Means Clustering method. Since we are writing the algorithm for directed network, I further modify the component functions, such as strength of ties, degree of centrality, profile similarity and fitness value. I then further implemented genetic algorithms (Mutation and Crossover) and its paralleling. Also, for paper, I have written sections relevant to my contribution, such as proposed framework.

7.2 Muneeb Shahid

Initially, I did the literature review and came up with the idea for this project. Then I was responsible for determining ways to design experiments for measuring the goodness of social circles and how to interpret them. Other than that, with the help of node2vec alogrithm I generated the node features for selected Slashdot signed dataset. Also, from implementation perspective, I have written functions for reading datasets, determining optimal number of clusters using K-means elbow method and calculating net objective values for features, Silhouette Coefficient index, Davies Bouldin index and Calinski Harbasz index. Also, for paper, I have written sections relevant to my contribution.

REFERENCES

[1] Viniti Agarwal and K.K. Bharadwaj. 2015. Predicting the dynamics of social circles in ego networks using pattern analysis and GA K-means clustering. In *WIREs Data Mining Knowl Discov 2015 (Volume 5, May/June 2015)*. John Wiley Sons, Ltd, New Delhi, India, 113–141. https://doi.org/10.1002/widm.1150

[2] Valerio Arnaboldi, Marco Conti, Andrea Passarella, and Robin I.M. Dunbar. 2017. Online Social Networks and information diffusion: The role of ego networks. *Online Social Networks and Media* 1 (2017), 44 – 55. https://doi.org/10.1016/j.osnem.2017.04.001

[3] Valerio Arnaboldi, Andrea Passarella, Maurizio Tesconi, and Davide Gazzè. 2011. Towards a Characterization of Egocentric Networks in Online Social Networks. In *On the Move to Meaningful Internet Systems: OTM 2011 Workshops*, Robert

Meersman, Tharam Dillon, and Pilar Herrero (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 524–533.

[4] Scott H. Burton and Christophe G. Giraud-Carrier. 2014. Discovering Social Circles in Directed Graphs. *ACM Transactions on Knowledge Discovery from Data* (2014), 27. https://doi.org/10.1145/2641759

[5] D. L. Davies and D. W. Bouldin. 1979. A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-1, 2 (1979), 224–227. https://doi.org/10.1109/TPAMI.1979.4766909

[6] Antonio Gonzalez-Pardoa, Jason J.Jungb, and David Camachoa. 2016. ACO-based clustering for Ego Network analysis. *Future Generation Computer Systems* (2016), 160–170. https://doi.org/10.1016/j.future.2016.06.033

[7] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable Feature Learning for Networks. SIGKDD. https://cs.stanford.edu/people/jure/pubs/node2vec-kdd16.pdf

[8] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Signed Networks in Social Media. 28th ACM Conference on Human Factors in Computing Systems. https://cs.stanford.edu/people/jure/pubs/triads-chi10.pdf

[9] Yanchi Liu, Zhongmou Li, Hui Xiong, Xuedong Gao, and Junjie Wu3. 2010. Understanding of Internal Clustering Validation Measures. IEEE 2010 International Conference on Data Mining. https://doi.org/10.1109/ICDM.2010.35

[10] Pranav Nerurkar, Madhav Chandane, and Sunil Bhirud. 2018. Understanding attribute and social circle correlation in social networks. *Turkish Journal of Electric Engineering Computer Sciences* (2018), 1228–1242. https://doi.org/10.3906/elk-1806-91

[11] Peter J. Rousseeuw. 1987. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* 20 (1987), 53 – 65. https://doi.org/10.1016/0377-0427(87)90125-7