Learning to Discover Social Circles in Ego Networks

Ralla Akhil and Garlapati Sai Snehith

***Abstract*— Our personal social networks are big and cluttered, and currently there is no good way to organize them. Social networking sites allow users to manually categorize their friends into social circles (e.g. circles on Google+, and lists on Facebook and Twitter), however they are laborious to construct and must be updated whenever a users network grows. We define a novel machine learning task of identi- fying users social circles. We pose the problem as a node clustering problem on a users ego-network, a network of connections between her friends. We develop a model for detecting circles that combines network structure as well as user profile information. For each circle we learn its members and the circle-specific user profile similarity metric. Modeling node membership to multiple circles allows us to detect overlapping as well as hierarchically nested circles. Experiments show that our model accurately identifies circles on a diverse set of data from Facebook,Google+, and Twitter for all of which we obtain hand-labeled ground-truth**

I. INTRODUCTION

Online social networks allow users to follow streams of posts generated by hundreds of their friends and acquaintances. Users friends generate overwhelming volumes of information and to cope with the information overload they need to organize their personal social networks. One of the main mechanisms for users of social networking sites to organize their networks and the content generated by them is to categorize their friends into what we refer to as social circles.

Currently, users in Facebook, Google+ and Twitter identify their circles either manually, or in a nave fashion by identifying friends sharing a common attribute. Neither approach is particularlysatisfactory: the former is time consuming and does not update automatically as a user adds more friends, while the latter fails to capture individual aspects of users communities, and may function poorly when profile information is missing or withheld.

In this paper we study the problem of automatically discovering users social circles. In particular, given a single user with her personal social network, our goal is to identify her circles, each of which is a subset of her friends. Circles are user-specific as each user organizes her personal network of friends independently of all other users to whom she is not connected. This means that we can formulate the problem of circle detection as a clustering problem on her ego-network, the network of friendships between her friends.

II. DATASETS

A. Facebook

This dataset consists of 'circles' (or 'friends lists') from Facebook. Facebook data was collected from survey participants using this Facebook app. The dataset includes node features (profiles), circles, and ego networks.

Facebook data has been anonymized by replacing the Facebook-internal ids for each user with a new value. Also, while feature vectors from this dataset have been provided, the interpretation of those features has been obscured. For instance, where the original dataset may have contained a feature "political=Democratic Party", the new data would simply contain "political=anonymized feature 1". Thus, using the anonymized data it is possible to determine whether two users have the same political affiliations, but not what their individual political affiliations represent.

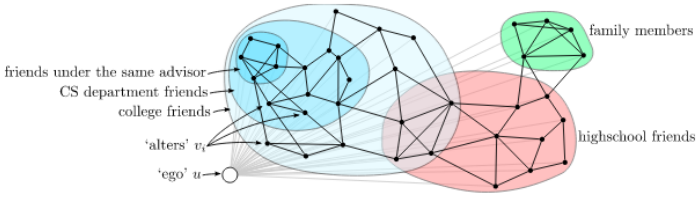

Fig. 1. An ego-network with labeled circles. This network shows typical behavior that we ob- serve in our data

In Figure 1 we are given a single user u and we form a network between her friends vi. We refer to the user u as the ego and to the nodes vi as alters. The task then is to identify the circles to which each alter vi belongs, as in Figure 1. In other words, the goal is to find nested as well as overlapping communities/clusters in us ego-network.

B. Google

This dataset consists of 'circles' from Google+. Google+ data was collected from users who had manually shared their circles using the 'share circle' feature. The dataset includes node features (profiles), circles, and ego networks.

C. Twitter

This dataset consists of 'circles' (or 'lists') from Twitter. Twitter data was crawled from public sources. The dataset includes node features (profiles), circles, and ego networks.

III. CLUSTERING ALGORITHMS

Clustering is the process of making a group of abstract objects into classes of similar objects. A cluster of data objects can be treated as one group. While doing cluster analysis, we first partition the set of data into groups based on data similarity and then assign the labels to the groups. The main advantage of clustering over classification is that, it is adaptable to changes and helps single out useful features

that distinguish different groups. In this problem our goal is to distinguish different groups based on the type of features. The different approaches for clustering are:

Partitioning Method: Suppose we are given a database of n objects and the partitioning method constructs k partition of data. Each partition will represent a cluster and k n. It means that it will classify the data into k groups, which satisfy the following requirements are Each group contains at least one object. Each object must belong to exactly one group. The major drawbacks in this method are : 1.We cannot fix the number of clusters. 2.Each node belongs to only one group. 3.The separated cluster may not have the common feature with the ego.

Density Based Method:This method is based on the notion of density. The basic idea is to continue growing the given cluster as long as the density in the neighborhood exceeds some threshold, i.e., for each data point within a given cluster, the radius of a given cluster has to contain at least a minimum number of points. The major drawbacks in this method are: 1.It is difficult to fix the threshold radius and density while using DBscan algorithm. 2.The separated cluster may not have the common feature with the ego.

Grid Based Method: In this, the objects together form a grid. The object space is quantized into finite number of cells that form a grid structure. Advantages are: The major advantage of this method is fast processing time. It is dependent only on the number of cells in each dimension in the quantized space. The major drawbacks in this method are: 1.The people in the same grid may not have common features. 2.Over fitting happens when trying to get perfect results.

Hierarchical Methods: This method creates a hierarchical decomposition of the given set of data objects. We can classify hierarchical methods on the basis of how the hierarchical decomposition is formed. As this method gives freedom to break at any level,we used this method to get good results. There are two approaches here

A. Agglomerative Approach

This approach is also known as the bottom-up approach. In this, we start with each object forming a separate group. It keeps on merging the objects or groups that are close to one another. It keep on doing so until all of the groups are merged into one or until the termination condition holds. The different metrics we used for merging the clusters are:

1) Minimum distance:Even though there is less similarities in the features of two clusters but based on the minimum distance between two objects they are merged,so it gives bad results.
2) Average Distance:Average distance between clusters also worked well.But since there are some anomalies,they are formed as separate clusters.The clusters thus formed are also approximately similar to given clusters.
3) Maximum Distance:When Maximum distance is used as a metric between clusters,the clusters thus formed are approximately similar to given clusters and better when compared to results of other methods.

We took feature vector for each alter(friend of an ego).Each feature vector represents to which school friend belongs, birthday, locality,etc compared with the ego.i.e.,if both ego and friend has same value then the value is 1 for that feature. We applied Agglomerative approach on this feature vectors. The number of clusters created is equal to the number of clusters given.

B. Divisive Approach

This approach is also known as the top-down approach. In this, we start with all of the objects in the same cluster. In the continuous iteration, a cluster is split up into smaller clusters. It is down until each object in one cluster or the termination condition holds. This method is rigid, i.e., once a merging or splitting is done, it can never be undone.

The feature vectors are same as described in agglomerative approach.In this we used connected components and created clusters for each connected components. Even in this approach maximum distance metric gave good results.

IV. Results

The created clusters thus formed by agglomerative approach are separated by features.The created clusters are approximately same when compared to the given clusters.