Detecting Clone Profiles in Social Media Networks

C. R. Liyanage

169318J

Master of Science in Information Technology

Faculty of Information Technology

University of Moratuwa

February 2019

Detecting Clone Profiles in Social Media Networks

C. R. Liyanage

169318J

Dissertation submitted to the Faculty of Information Technology, University of Moratuwa, Sri Lanka for the partial fulfillment of the requirements of Degree of Master of Science in Information Technology

February 2019

Declaration

I declare that this thesis is my own work and has not been submitted in any form for another degree or diploma at any university or other institution of tertiary education. Information derived from the published or unpublished work of others has been acknowledged in the text and a list of references is given.

Name of Student

…….

Signature of Student

……..……

Date:

Supervised by

Name of Supervisor

…….

Signature of Supervisor

……..……

Date:

Acknowledgement

I wish to express sincere appreciation to,

My research supervisor, Mr. S. C. Premarathne for dedicating his valuable time for guiding me throughout this.

My husband and parents for providing me with continuous encouragement throughout my years of study.

Finally, for my colleagues for their cooperation and support.

Thank you.

Abstract

With the popularity of Online Social Networks (OSN), the number of different types of digital attacks has been increased causing lots of damages to their users. Identity Clone Attack (ICA) is one of the leading among them which illegally uses the information of a genuine user by duplicating them in another fake profile. These attacks severely affect a true and innocent identity since it can be misused by another malicious profile. Hence these clone profiles need to be identified and removed in order to increase the protection of users. Many researchers have tried to solve the problem of clone profiles in OSN, however more robust solutions are still to be taken. This study introduces a model to detect clone profiles on Facebook by clustering based on weighted categorical attributes and estimating the strength of friend relationship among friends. The list of possible clones with the amount of clone percentages to a given victim profile was presented as the output of the model. With the use of Agglomerative hierarchical clustering algorithm and Jaccard similarity measurement, a low average within cluster distance and a precision of 88.75% has shown in the results

Table of Content

List of Tables

List of Figures

List of Equations

Chapter 1

Introduction to fake and clone profiles

1.1. Introduction to the background

Recently Online Social Networks(OSNs) have become a significant part of people life where 2.46 billion of the global population is using it and expected to reach around 2.95 billion in 2020 [1]. Among various social platforms such as You Tube, WhatsApp, Facebook, Instagram, Twitter, Google+ and LinkedIn, the Facebook remains as the world most popular social network as of September 2017 [1]. These networks have facilitated lot of benefits to its users such as to keep contacts with their friends, allow them to find news and updates around the world, provide business opportunities, share ideas and knowledge. Moreover, social media has changed the way of people interact with each other and users have tended to expose their public and private information on such platforms. However, with this rapid growth and wide usage, OSNs have led to some negative outcomes as well. Risks of fraud and identity theft are two of most popular issues that can be found in OSNs, and these problems are generated through fake profiles. According to statistical estimations Facebook has 81million of fake accounts whereas 5 percent of Twitter accounts are forged [2].

1.2. Profile Cloning Environment

Identity theft or Identity Clone Attack (ICA) is one of the most popular attacks in OSN and it is performed by profile cloning. Profile cloning is a way of stealing information from an existing user and creating new similar fake profiles using those details. Cloning a profile on OSN can be done with several intensions such as to trick users, abuse financially, damage a person's reputation and to steal sensitive data of others[3].

When cloning profiles in OSN, adversary first creates a fake profile using the publicly available attribute information of the victim profile. A profile in social network platform has a name, most probably a first and last name with other set of attributes such as birthday, hometown and school to represent its identity. In profile cloning, most of the attributes in the victim profile will be copied by the clone profile. Usually the name is the main feature of both clone and victim should have in common[4]. However, some of the attributes will not be copied the same value rather some will be

kept as empty or private. This is because clones can duplicate victim's features as well as they can maintain their own privacy setting by making some of the attributes private. In addition, an adversary can make some attributes public in which the corresponding victim had set to private such as birthday where most of the users try to keep it private. According to the study[4] these activities may make the faked identity more realistic.

Typically, after cloning a profile it will send friend requests to friends of the victim. At this stage, since the clone profile looks more similar to the genuine profile, friends of the victim will tend to accept a friend request from the clone without noticing that it is a duplicate profile of their friend[3], [4]. Hence, adversary gets the chance to publish misleading contents to the victim's friend audience using clone profile in order to damage his good profile. In addition, there can be some other problems caused due to the exposition of victim's friends' private data to the adversary.

Before adding a friend to the network, a cautious user will first look for his friend list to check whether that user already exists or not. In that case, adding a considerable number of friends of the victim may not be easy. Hence the adversary tries to add the recommended friends of the victim so that the clone becomes more genuine and makes it difficult for the victim to add those recommended friends[4]. The recommended friend list is usually generated by the OSN platform. They are the list of people who are not yet friends of the victim but having similar backgrounds or mutual friends between them.

As mentioned above now a clone profile and the genuine profile will be very similar to each other in terms of public attribute values, friend networks and recommended friend networks. Under these assumptions the purpose of this study is to introduce a novel detection model that can use similarities in profile attributes and network details to find the possible clones for a given victim. In order to increase the efficiency, the initial search space will be reduced in a larger amount by filtering only the profiles with names similar to the victim's name. Next, these filtered profiles will go through clustering based on public attributes and filtering using network similarities. Finally, suspect profile list will be presented with the amount of duplicability as a percentage. The model was developed for Facebook which has the highest popularity, largest number of user profiles and also with highest number of fake profiles[1], [2].

1.3. A brief from the previous work

As mentioned above, there are many different anomalies and security issues for the people who are using these media and those problems have taken the attention of research communities to find detection and protection solutions. There are different existing solutions to resolve this problem of detecting fake profiles in OSNs using different approaches. In paper [5] and [6] the author has tried to identify fake profiles based on classification techniques. Some algorithms have used graph based approaches, for an example in [7], has introduced a detection mechanism called Fake Profiles Recognizer (FPR) which recognize his trusted friends by representing profiles using a mathematical pattern as a regular expression. Another case study [8] has used a social graph to represent profiles as nodes and relationships as edges to identify their interactions. The literature has introduced some other approaches to specifically handle the problem of profile cloning in OSNs. In [9] the authors present a way to detect duplicate profiles over different social platforms using binary classification. Some approaches are there to identify duplicate profiles by finding similarities in attributes and behaviors between profiles [10]. According to the details gained, finding a solution for the problem of fake identity detection is crucial and it is a complex task due to the dynamic characteristics of the online social network environment. Nevertheless, due to limited access to bulk of diverse OSN data, the researchers have faced difficulties of conducting good researches.

1.4. Problem Statement

Due to the severity of the damages cause through this Identity Clone Attacks to OSN users, the need of a good detection methodology is of vital importance. Though previous researchers have presented different types of solutions to detect ICA attacks, due to many reasons including complexities and less security concerns, still there is a need of more reliable and efficient solutions.

1.5. Aim and Objectives

The main aim of this study is to minimize anomalies related to identity stealing and increase the security and privacy conditions by detecting duplicate profiles in a selected social media platform, which is the Facebook. To fulfill that aim there are several sub objectives such as,

- Identify profiles with same names as the name of the victim/ genuine profile
- Find the similarities between the victim and clone candidates based on interested attribute features and network connections
- Based on the similarity results, state the candidate profiles as duplicates of the victim or not
- Validate the result using known clone profiles

1.6. Research Process, Inputs, Outputs and Beneficiaries in brief

The experiment proposed in this study has focused on detecting clone profiles specifically in Facebook. Along with the highest popularity and largest number of user profiles of Facebook network, the degree of profile cloning has become increased and so the necessity of research solutions has also increased.

This research work finds the duplicate profiles for a given genuine profile using their public attributes and friend network details. As the output, a percentage for each suspect profile will be generated and that value indicates the similarity between the that profile and given genuine profile. This novel approach can be used to detect duplicates of any profile; hence this is useful for all the users in OSN to be protected from identity clone attacks and to increase their trust on the network.

1.7. Summary of the first chapter and heading to next chapters

The rest of the report has organized as follows. Chapter 2 provides the details about the previous work done to resolve the problem of clone profile detection. Then in Chapter 3, the technologies adapted in the research work will be discussed. The novel approach and the use of technologies will be presented in Chapter 4. Under the next chapter, the overall design details of the introduced work will be presented. In here, a model with the research steps for detecting clone profiles, and how these steps individually can be implemented to get the final results will be reported. In Chapter 6, the explanation of the implementation and experiment has presented by mapping the proposed design with the actual work done. Later in Chapter 7, a discussion will be carried to evaluate the output result with expected results to check whether the initial objectives have been achieved. Finally, in Chapter 8, the conclusion of the research work will be summarized with the findings, limitations and future improvements.

Chapter 2

Current state of detecting Duplicate Profiles in OSN

2.1. Introduction to the background

Researchers have addressed fake profiles in two aspects either as a duplicate for a specific existing account (profile cloning) or as a new profile with random details. Profile cloning again tested across different platforms which made the security of social network more robust. They have selected different social networks and most common selections were Facebook [9], Twitter, Google+ [11] and LinkedIn, where the user profile attributes and behaviors are significantly different one to another.

Study [9] proposes a three step model to match two different profiles from different social media platforms. They have used a binary classifier for feature extraction based on users' information regarding friend requests and friend lists. This method presents a more robust model by using a string-matching similarity algorithm to find profile attribute similarities. However, they have not tested their algorithm using a real dataset. Hence the accuracy and effectiveness of the output is questionable. The authors in [10] have compared the impact of different parameters on verifying the results of the outcomes. First, they have selected the victim and then found list of potential clone profiles. By comparing clones with victims, they have finally verified the results as which profiles are clones.

The study [12] has tried to find clones in social media where the concept was evaluated on users' original profile data to catch similar accounts across OSNs. According to the detected profile similarities, a similarity score has been calculated based on common values of information field and profile picture. Another study [11] for detecting duplicate profiles in OSNs has performed and they have considered more similar steps as previous cases [12]. First, they extract information from users' profile such as birthday, age, education, work place and then extract information from profiles with same names. Finally, they have calculated a similarity index of all the profiles found. Most of the studies have built their approaches based on attribute

similarity models. In paper [13] also have done the same thing but further they have considered about a friend network similarity value.

2.2. Current Situation

The area of research for detecting duplicate profiles in online social media networks has evolved recently and most of the research findings were published after 2010. Since the research approaches are different from each other depending on different OSNs, selecting the most suitable platform was the first most important step. After that finding data sets with interested features, applying suitable methodologies and evaluation of results must be done accordingly. Current background of this research area will be discussed in this section.

2.3. Platform Selection

Single site and cross site profile cloning are two types of cloning attacks where in first type creates an account of the victim in the same social network and sends friend requests to victims' friends whereas in cross site creates an account of victim in a new network and sends requests to friends who are in both networks [14][3][11]. According to these two types researchers have developed their fake profile detection algorithms on either specific network or across multiple networks [15]. In present as Facebook is the most popular OSN, many researches have selected it as the platform for their research work [5][6][8]. Not only that, some authors have used multiple platforms such as Google+ and Twitter along with the Facebook as their social environments [7][16].

2.4. Data Collection

In each profile in OSN provides lots of qualitative and quantitative information such as gender, location, education, work, age, number of friends, comments, likes, images etc. However these information provide different accessibilities for different audiences since some are public and others are private [5]. In many researches public data has been used due to limitations of gathering private data of profiles [15][17].

However in [9] the author has not used real data set for his implementation. Data gathering has mainly carried out in several ways where creating experimental fake profiles or called as “Honey profiles” has done by [8] and this method was better than the way of data gathering via APIs since researchers can gain data by controlling the conditions as they want. They have created several honey profiles with different features and collected data once each day for one month. However, this method has limitations when considering vast amount of data collections.

Some researchers have collected real profile information using Facebook Graph API along with Python [7][5] and fake profile dataset has provided by Barracuda Labs [5]. Some have scrapped data from friend accounts and for that they have implemented an anti-scrap detection technique to prevent Facebook from detecting it [5]. Paper [7] has used fixed number of profiles around 3000 and these were downloaded from Stanford Network Analysis Platform (SNAP Library). They have divided the dataset into two parts one half as real profiles and other as fake profiles. Another study [6] has collected their initial data set of 4.4million public posts using post search API of Facebook. Social Snapshot tool developed by Huber is one of the tools used in [8] to collect Facebook user data. Researchers in [20] have collected data for their classification methods by collecting data manually from their own networks (author’s networks). They have collected 17 features from 4708 of users and identified different types of users as fakes, real, assumed to be fake or real. They have manually detected 230 profiles as fake by identifying their suspicious behavior of spreading lots of promotions and spam contents. After identifying users, they have used Facebook API to collect user-feeds through python wrappers. All activities on feeds are captured as JSON objects.

2.5. Existing Approaches

2.5.1. Using Classification Algorithms

Some algorithms have tried to solve this problem of identifying OSN fake profiles based on classification approaches. In [5] the author has used three classification algorithms, Support Vector Machine (SVM), Naive Bayes and Decision trees and have compared the efficiency among each. After selecting the profile to be tested they have extracted the required features (Gender, Number of friends, education and work,

relationship status, numbers of photos tagged, number of uploaded photos etc.) and then using the classifier determined whether the profile is fake or not. Then again, the result has used to train the classifier in order to obtain more accurate predictions. According to the results SVM has selected as the best classification model where Naïve Bayes has given the lowest performance. Another research study [6] has conducted to find malicious Facebook pages using Artificial Neural Networks. The set of words in published contents has used to differentiate malicious and true pages.

Some approaches are there to find user profiles belong to the same user over different social networks [17]. They have generated a similarity vector using known dataset of paired accounts belongs to the same user across multiple networks. Then these vectors were used as the training dataset for supervised classifiers such as KNN, Naïve Bayes, Decision trees and SVM. However this approach is using more static attributes (Name, Location, Description, Profile image and Number of connections) when considering similarity vector whereas in some approaches use more dynamic behavioral features like in [16] which have shown more robust and accurate results.

2.5.2. Social Graph based approach

In paper [7] the author introduces a detection mechanism called Fake Profiles Recognizer (FPR) which authenticate and recognize his trusted friends as well as detect Fake ones by modeling the online social network graph by representing the identity of each user by a Friend Pattern. A profile will be a fake to a selected profile, if it has indicated by a fake instance which came from another friend pattern and will not accepted by the friend pattern processor. This friend pattern has used to distinguish duplicate profiles in OSN. This approach has proved higher accuracy than SVM [5] and lower F-Measure values than Naïve Bayes approaches [5]. However, in case of lesser number of fake profiles this algorithm has unable to recognize the fake profiles. A case study [8] has performed by illustrating its friendship network using graphs where nodes represented profiles and friendships among profiles represented edges. They have presented some concepts, network density, degree of nodes, and correlation between nodes in the process of identification fake nodes. Finally, they have concluded the profiles with a smaller number of activities and high number of friend have more chance to be fakes.

The approach [3] has evaluated the identity of clone profiles in the same network using two concepts in which the second one is based on its' strength of the relationship measures. For this social network data were modeled using a weighted graph and they have tried to consider user interactions not only based on friend requests rather considered more linkage between profiles such as active friends, page likes, URLs, friendship graph and mutual friends' graph.

In [18] a novel social graph topology called "Trusted Social Graph (TSG)" has introduced by using a special type of graph called "DeBruijn graph" to visualize the trusted instances within the social network. They have analyzed the social profiles by evaluating their friend patterns using mathematical expressions. Finally, the incoming instances were checked against the model and decided whether that profile is fake or real.

Some algorithms like [19] have presented a method to detect clone profiles using a graph and network based approach by analyzing the structural similarity of the social network. The authors have first selected a node to analyze from an analyzed network and get nearest neighbors considered node. After measuring the similarity of nodes, it will detect duplicate profiles as gave highest frequency of attribute similarities. Furthermore, due to the usage of k-nearest neighbor algorithm, this approach was able to recover hidden values of attributes of user profiles.

2.5.3. Matching similarity attributes

In study [15] the similarity of two profiles has checked based on their HTML structures. They have conducted techniques on exact matching of attributes to match usernames by doing string comparisons and partial matching of related attributes to match parts of profile attributes such as location and address. In [9] has also used a similarity matching algorithm but it has shown higher results due to its recursive matching technique. As mentioned under graph based approach, the study [3] has evaluated the profile identity using two concepts which the first one was based on calculating profile similarity using selected attributes, the first name, family name and location. After filtering suspicious accounts based on these attributes' similarities, they are evaluating the strength of relations and finally have identified the fakes. The literature has introduced another approach [10] to detect profile clones by comparing

five different similarity measures which includes two more additional attributes, gender and education details than given in study [3]. However, this study has used a limited dataset for their developments.

The methods like [11] have calculated a similarity index after comparing the original profile and other searched accounts. They have assumed if the similarity index is high the profiles may be cloned. However, the other assumption they have made as the fake profiles will give lowest similarities is not acceptable since there can be profiles with less similarities to each other but still real. The approach [13] has introduced a weighted dice similarity measurement to calculate the similarities and rank the selected attributes. They have assigned weights according to the importance of each attribute for each person. This method can give more reliable results since the importance of attributes may vary from person to person. Some algorithms [12] have directly matched the strings in information fields to measure the similarities between profiles. However, in case of incorrectly typed information this method will give inaccurate results. Same as most of the approaches, the paper [20] has also discussed about an attribute similarity and friend network similarity approach. They have considered three types of friend network features for analysis, friend list, recommended friend list and excluded friend list. Furthermore the study [8] has focused more on analyzing the location based attributes such as work and educational places and current locations and has found they will give stronger factors in fake identifications. Not only that, they have used 12 classifiers for detecting task of the fake profiles. The study [28] has taken age, location, gender, and user interests as the attribute set and they have showed better results on those.

2.5.4. Analyzing user behavior changes

According to [14] the interested features can be categorized into two as behavioral and non-behavioral attributes. Due to the anomalous behavior of fake profiles they are easy to identify by analyzing behavioral patterns [16]. Paper [6] has used a bag-of-words collected from recent activities of Facebook pages and extracted patterns from them. Furthermore, they have analyzed the behavior changes in such pages. The approach [16] has used a combination of statistical models and sudden behavioral changes in user profiles to detect fakes. They have considered detecting only the

malicious behavioral changes for their algorithms since users can experience sudden changes in their behaviors due to many other legal reasons as well. In [21] the authors have used a text mining approach to measure the similarity between text information such as posts and comments on two types of social media public pages. As mentioned in [20] the researchers have used 17 dynamic profile features to evaluate the similarities between users such as, average posts like received, average posts liked, average post comments received, average tags etc. to analyze the user activities and interactions with each other.

2.5.5. Validation

Finally, most of the OSN researchers were unable to validate their results on a real platform while some others[13] have performed result validation through social authentication in which asking general questions from the suspect clones about their profile friends' information. When these suspects are unable to answer the questions, they will be verified as clones. Another way of validation is asking for unique real-world ID from the suspects[22]. Furthermore, the researchers of some studies[23] have got the help of Facebook security team to validate their findings.

Chapter 3

Technologies used for Clone Detection Process

3.1. MySQL

MySQL has used to create tables and to change the data into required formats. phpMyAdmin is using as the tool of MySQL since it is free and open source. After the initial alterations data files were saved into CSV format which was easy to read from Java language and to load into RapidMiner software for further processing.

3.2. Java

The main advantage of using java as the development language is that it's platform independency which enables to run the source code in any computer system and to move from one to another. Moreover, RapidMiner application is also developed using Java, therefore, use of Java for accessing the tool is easier than using other languages. This research study has used java to calculate the necessary measurement values related to attributes and network data of profiles and to build larger data structures to store data.

3.3. RapidMiner Studio 9.1

RapidMiner is a powerful data analytical tool that can use to perform data preprocessing, regression, association mining, clustering, classification and visualization etc. There is a commercial version with lot more functionalities, however the free version was used for this research study. See the main window in figure 3.1

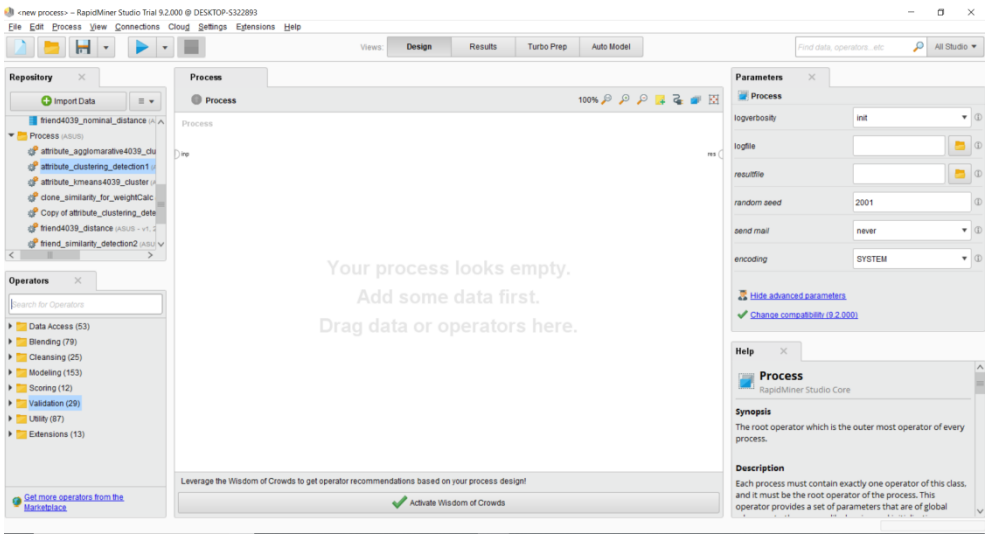

Figure3.1: RapidMiner Studio 9.1

In this study, the initially preprocessed data which were taken from the MySQL were stored for further preprocessing in RapidMiner. The preprocessed data along with the calculated measurements have used in RapidMiner to extract similar profile information by clustering and finding data similarities.

3.4. Datamining through Clustering

Clustering is a technique that use to group similar objects into same clusters and dissimilar objects to other clusters. It uses to extract information from unlabeled data. In this study, several clustering algorithms such as, Kmeans, Kmedoids and Agglomerative, were tested on the data to find the best matching one. According to the type of the attributes the distance or the similarity function has been selected.

Agglomerative clustering is one of the common hierarchical clustering algorithms and by flattening the cluster result at a given level, each element can be assigned exactly into one single cluster. Hierarchical clustering performs well on categorical data than numerical.

Both Kmeans and Kmedoids are centroid based clustering algorithms and they suits more on numerical than on categorical data. However recently there were many researches who have introduced many distance measuring algorithms on categorical data and many have experimented the application of Kmeans and Kmedoids algorithm on both mixed numerical and categorical data.

3.5. Similarity measurement algorithms

Similarity measurements need to be used in both clustering and friend network similarity calculation. Those measurement will indicate how much similar the given two objects are. To find the similarity between nominal attributes of two profiles, the Jaccard similarity was used. More details on Jaccard similarity will be discussed under chapter 5.

3.6. Hardware environment

The hardware environment under which the research work carried out had RAM of 8GB, a Core i5 processor version and a hard disk storage of 1TB.

3.7. Advanced Data Generator

Advanced Data Generator is a tool which was used to generate the artificial data set for the testing set of the model. The output data were directly saved into MySQL tables. See the home window of the tool in figure 3.2

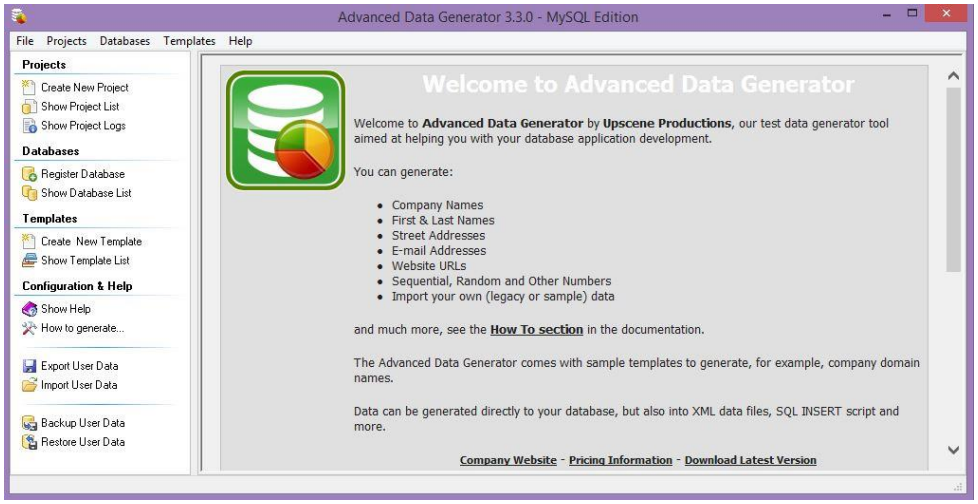

Figure 3.2: Advanced Data Generator tool

Chapter 4

A Novel Approach for Duplicate Profile Detection

4.1. Introduction

In this chapter the introduction to the novel approach will be discussed. The main idea of solving this problem is based on finding similarity of users through their profile attributes and friend connection information. As stated in chapter 2, there are many different approaches followed to solve the similar problem, however still no one could present a perfect solution due to the varying nature of the online social media platforms.

This chapter highlights the key features of how this novel approach is distinguish with the existing approaches to detect clone profiles in online social media platforms. Nevertheless, this chapter will discuss how the technologies states in previous chapter are used to do the actual work.

4.2. Inputs to the model

User profile information related to their attributes and friend network details have taken from an online data repository. A considerable effort has taken to get these data into the required formats and this was a challenging task due to the large amount of them.

4.3. Outputs from the model

This model has generated a threshold similarity value as the measure of filtering the genuine profile from the clone ones. Finally based on this threshold similarity, the model states the percentage of the possibility of a profile that can be cloned to another genuine profile.

4.4. Process in brief

In this research, there are three main stages of matching a given victim user profile with the other user profiles to detect the possible clones. In the first stage, the possible candidate profile set has filtered based on the profile name of the victim. Then in the second stage these candidate profile set has sent through a clustering approach based on their other attributes. Finally, the profiles fallen into the same cluster as of the

claimed victim's, have forwarded to compare with the victim's similarity values of mutual and recommended friend networks. According to the results a final discussion has carried out to find the accuracy of the model and to verify whether the expected results have been achieved.

4.5. Platform Selection

Currently lots of different Platforms are available as Online Social Networks and some of them are very popular among people around worldwide. Due to user friendliness, attractive features and easiness of using, Facebook ranks the top among others. However, with the extremely large number of users and less security features Facebook is facing the problem of identity frauds in a severe way. For an example, when user creates an account in Facebook, they must go through only little information filling and one user can have several accounts. With this easiness of creating user accounts, Facebook was unable to prevent duplicate profile creation. By considering these facts the present study decided to build the model on Facebook as the OSN platform.

4.6. Data Collection

A good collection of data is the most important factor of a successful research. Here in order to detect cloned fake profiles in OSN, which is a very large network consists of billions of variety users with distinct characteristics, the data input should be extremely realistic and relevant. However due to security and privacy concerns, social networks maintain restricted access to others private data. Hence most of the researchers are suffering from the deficiency of good data set with variety of features which can be used to easily analyze user behaviors. As literature states there are many tools which can use to get real user data from different platforms. However, when comparing to social networks like twitter, Google+ and LinkedIn etc. Facebook is the most restricted data provider and it allows only getting public data of the users. Under this condition use of an API to get large amount of data from Facebook with all the interested features was a difficult task. Hence a dataset with considerable number of attributes and details of friend connections was downloaded from the online data repository of Stanford University (SNAP)[24] .

SNAP has collected the data for their research from a survey. This dataset includes set of integer IDs (Nodes) which represent users in the Facebook Network, friend

connections between those users, profile features (attributes), circles, and ego networks. Since this research is focused on friend connections and attributes of a profile, only the relevant data is used for further processing. Furthermore, Facebook connections are undirected in the sense if node n1 is a friend of node n2, then node n2 will be a friend of n1 also. In the feature dimensions, features are '1' if the user has that property in their public profile, and '0' otherwise. This feature data has been anonymized for Facebook users, since the names of the features would reveal private data [24]. Bellow gives some of the statistics about the dataset.

- Total number of nodes : 4039
- Total number of connections : 88234
- Number of triangles : 1612010
- Diameter (longest shortest path) : 8
- Number of profile features : 26

Additional to the data downloaded, some other relevant data such as clone profile information, recommended friend list and testing profile set needed to be generated artificially.

4.7. Performing clustering

Clustering has performed as the datamining technique in order to detect natural groups among the candidate user profile set, and it was needed by the model to divide the users into similar groups based on their attributes. To accomplish this task, clustering was used on nominal profile attributes such as gender, hometown, birthday, school, work location etc. For that, the RapidMiner tool has used with an appropriate clustering algorithm and a similarity measurement.

In generally, however the evaluation of clustering algorithms is difficult because the success of clustering is subjective to the requirements of the model and the used algorithms and measurements. There is no well-defined metric for clustering as in classification. Hence in order to train the model, a known clone label set has used without adding them into the cluster process. After clustering it helped to verify the results.

Chapter 5

Clone Profile Detection - Analysis & Design

5.1. Introduction

This chapter will discuss the model design and how its steps declared as modules in detail. In overall, this model will work on Facebook user profiles which consist of attribute and friend network information and detect the possible duplicate profiles for a given victim user profile. This possibility is given as a percentage based on a pre-estimated threshold value using the similarity between the victim and the suspect clone. The model design is stated in a greater detail below.

5.2. Overview of the Proposed Model Design

There are two main stages of the model development. First but the most crucial step was the data preprocessing. Selecting the most relevant attributes, filling missing values, getting the data into required formats and files and modifying a selected set of profiles as the clone profiles were some of the things done during the preprocessing. The next stage was the detection process and it was divided into several sub steps. Not all the profiles will go through all these steps but only the profiles filtered by each step will be forwarded to next steps.

First the model will input the name or the ID of a victim profile who has claimed to find his clone profile/s and using this, other profiles with the same name as the victim will be filtered and forwarded to the next detection step. These selected set of profiles are referred as the candidate profile set. It is assumed that the first step of making duplicate profiles is stealing a name of another genuine identity since the name is the main feature used to recognize a person. When a fake user wants to forge a genuine user, it is assumed that it will make the profile looks similar to that user. Hence most of the public features will be same in both profiles.

Under these circumstances the profiles with the same name including the victim will be sent to the next step which is the clustering based on their rest of the attributes other than the name. According to the cluster results the candidate profiles grouped into the same cluster with the victim will be sent to the next step of detection as the suspect list. By now the filtered profiles are having a higher similarity to the given

victim based on their attribute features. The Figure 5.1 shows the steps of the detection stage.

The next step will further verify the duplicability by checking the friend and recommended friend network similarities between the victim and filtered suspect user profiles. If this calculated profile similarity value between each pair of victim and suspect is above a predefined threshold, then these profiles will be selected as possible clone accounts of the victim account and this possibility is given as a percentage.

The threshold value will be calculated by using the output, the possible clone profile set of the training phase and the known clone set. In the testing phase an artificially generated profile set has used to validate the system to confirm the accuracy of the model.

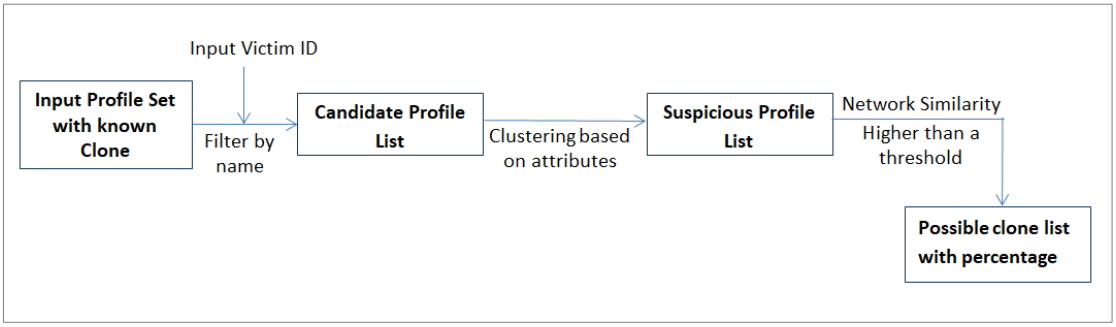

Figure 5.1:Proposed steps of detection stage

5.3. Attribute feature selection

The key goal of an adversary is to add friends of a genuine user by duplicating the attribute features of that account so that it can perform misleading activities using the name of the genuine user. In this type of clone attacks the first thing a fake profile will do is obtaining victim's personal information from his online profile or some other ways. In Facebook this personal information normally contains, first name, last name, location, occupation [24] etc.

Since there are lots of attributes related to the Facebook profile and all of them are not having equally importance in finding similarities between profiles, this study has chosen some of the features according to a justification from the literature as frequently used attributes in detection methodologies [3], [10], [13], [17], [22], [25], [26]. Most of the researches have tried to build their detection methodology based on analyzing the values of this kind of static attributes and those were described under chapter 2.5.3 in detail. In the downloaded dataset there were 26 attribute features

available and the values of them are replaced with a string “anonymized feature1” according to the uniqueness of the value. However, lots of preprocessing has done to modify the initial formats of the data sets into new formats required by the computations. These preprocessing steps will be discussed in next chapter. According to the justification, the most important 10 features were selected among the 26 and those are listed on the following Table 5.1.

No	Primary Features	Secondary Features	Selected Attributes
1	first_name		first_name
2	middle_name		
3	last_name		last_name
4	name		
5	birthday		birthday
6	gender		gender
7	religion		
8	locale		
9	location		location
10	hometown		hometown
11	languages		
12	education	Type	
13		School	School
14		Degree	
15		Classes	
16		concentration	
17		With	
18		Year	
19	work	Employer	Employer
20		Position	Position
21		Location	Location
22		start_date	
23		end_date	
24		With	
25		Projects	
26		From	

Table 5.1: Profile attributes/ features considering

5.4. Artificial Clone Profile Set Generation

Due to the difficulty of finding an originally verified clone profile set, this research study modified some of the existing profiles in the dataset as the clone set which is to be 2% of the original dataset and it was around 80 profiles. According to the characteristics stated in Chapter 1, clone profiles were given the same name of the victim, similar values for many attributes and few NULL values. Figure 5.2 shows an example of artificially modified clone and victim pair in term of their attributes

user	birthday	education_school	first_name	last_name	gender	hometown	location	work_employer	work_location	work_position
126	NULL	50	1084	599	77	1757	132	NULL	1657	NULL
518	NULL	50	1084	599	77	1757	NULL	NULL	NULL	NULL

Figure 5:2 Example attribute set of victim and clone profile (victim top, clone bottom)

It is known that a clone will not only duplicate a victim's attributes rather it will have similar network details due to the addition of same set of users. Thus, the friend networks of the victims were also modified in order to be similar (not exactly) to the network of the victims. Furthermore, the dataset was created in a way that one genuine user can have one to three corresponding clone profiles.

5.5. Filter candidates by name

All the users in the data set are represented by a numerical id which is from 0 to 4038 and each user has a feature vector which represents the availability and the anonymized values of each user. It was assumed that all the users in the model are having a first name and most of them are having a last name also. As the first step of the detection process the name of all the user profiles were compared with the name of the given victim. If two profiles have the same anonymized feature given for the name, Example: anonymized feature12, then those profile set will be sent into the candidate pool. The feature vectors of set of filtered users will be evaluated further.

5.6. Clustering based on attribute features

Many people on online social media network can have similar names with default privacy settings. Of course, then they appeared to be very similar to each other since

name is the first thing a user will check when they need to identify a user other than the profile picture. Hence, a clone profile detection algorithm may need to consider lot more other attributes than just the names to avoid the assumption that profiles with similar names are considered as fakes. According to this necessity, after filtering the candidate list based on similar names, the next step is to consider the rest of the attributes to cluster users.

Prior to the process of clustering the selected attributes were assigned with a weight according to the importance of them. Weights reflect the effect of each attribute during the process of detection and decision-making. The study [11] has used rank-sum weighting formula to calculate the weights for the attributes after ranking them according to the order of importance. Nevertheless, the previous studies have presented some other formulas such as rank exponent, rank order centroid, rank reciprocal etc.

However, in this study the weight calculation was done using a simple but effective method represented by the study [24]. This finds the similarity, means that two values given under same attribute of the known clone and victim are same or not. After finding the entire similarities among pair of victim and clone, the average was taken for that attribute. In the same after finding the average similarities for each attribute, then those were used as the weights of the attributes. In other word, when the average similarity of an attribute is high, then the weight assigned to that attribute is high. The table 5.2 shows the process of estimating the weights briefly.

User	Attribute1	Similarity	Attribute2	similarity	Att3	Similarity
Victim1	315	0	763	1		
Clone1	2103		763			
Victim2	410	0	103	1		
Clone2	NULL		103			
Victim3	26	1	56	0		
Clone3	26		89			
Weight	1/3=0.33		2/3=0.66			

Table5.2: Attribute Weight Calculation

Many victim clone pairs have similar values for attribute 2 than of attribute 1. Hence it can be stated that in the detection process attribute 2 has more importance than the attribute1 so a higher weight is assigned to that.

After estimating the weight values for the attributes, the next step was to find a best clustering algorithm and best number of clusters.

The best number of clusters (K) considering the density performance for several clustering algorithms, namely kMeans, kMedoids and Agglomerative were calculated using the filtered candidate lists of each victim of the dataset. Then the average of the number of clusters for each of these clustering algorithms was found as given in below figure 5.3. This task was performed using the Rapidminer Optimize Parameters (Grid) Operator.

Average Number of Clusters (K) with Density Performance		
KMeans	KMedoids	Agglomerative
7	6	6

Figure 5.3: Clustering Algorithms with number of clusters

The same optimization operator in Rapidminer was used to find the suitable clustering algorithm among Kmeans, Kmedoids and Agglomerative. Due to the highest distribution performance shown as in figure 5.4 the Agglomerative clustering with complete Link Distance and corresponding K value was selected to cluster the profiles using nominal distance.

Average Distribution Performances		
KMeans	KMedoids	Agglomerative
0.443	0.43	0.526

Figure 5.4: Clustering Algorithms with their distribution performances

5.7. Network Similarity Calculation

Similarity is the measure of how much alike two data objects are. Profile similarity measurement is a value calculated to evaluate whether a given profile has a possibility to become a clone of another account based on their networks. If the network

similarity is higher than a predefined threshold value, then one of the two profiles is said be cloned. To calculate the similarities there are several similarity equations such as dice similarity, Jaccard similarity, overlap similarity, and cosine similarity etc.[13]. Among them Jaccard similarity and cosine similarity are two very common measurements used. However Cosine similarity is popular for comparing two real-valued vectors, but Jaccard similarity is common for comparing two binary vectors [27]. Hence here the Jaccard similarity (Equation 5.1) is used as the similarity measurement to calculate how given two profiles are matched to each other and it will be value between 0 and 1.

$$J(A,B) = \frac{A \cap B}{A \cup B} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Equation 5.1: Jaccard Similarity Measurement

This research work considers both friend network similarity and recommended friend network similarity to calculate the profile similarity measure.

In order to find the network similarity between two profiles some of the past researches have considered only the similarity between the friend lists of two profiles [9][10], while some have considered an additional network information derived from recommended friend list and Excluded friend list of victim profile[13][4]. In this research friend network and recommended network information were considered to calculate network similarity among two profiles. However, only the friend network information of the users were available in the data set. Hence due to the unavailability of an actual dataset, recommended friend list has added to the data set manually. Recommended friend list for a victim was not randomly selected from their non-friend profile list. Rather, a set of non-friend users with higher number of mutual friends were selected as recommended friends of a particular user when they have same values on attributes such as hometown, location, school and work employer[10], [28].

5.7.1 Friend Network

When cloning a profile in OSN the first thing the adversary will do is duplicating the attribute values of the victim profile. After that to further increase the profile similarity, adversary tries to duplicate the friend list of the victim. At this stage since the clone profile looks more similar to the genuine profile, friends of the victim will tends to accept a friend request from the clone without noticing that it is a duplicate profile of their friend [4].

Based on the Jaccard similarity measurement following Equation 5.2 can be used to measure the similarity between friend lists (F) of two profiles.

$$S_{ff}\ (Pc, Pv) = \frac{Fc \cap Fv}{Fc \cup Fv}$$

Equation 5.2: Friend Network Similarity based on friend lists of two profiles

S_{ff} – Similarity between friend lists of two profiles

Fc – Friend List of Clone Profile

Fv – Friend List of Victim Profile

$Fc \cap Fv$ – Common friends between the profiles of clone and victim (Mutual Friends)

$Fc \cup Fv$ – Total friends available in the clone and victim networks

Example1

Friend	User 0	User 1
User 0	0	1
User 1	1	0
User 2	1	0
User 3	0	0
User 4	1	1

Figure 5.5: Friends of two users

When friends are summarized in an adjacency matrix it looks as follows. Using Jaccard index the friend network similarity of two public profiles can be calculated as,

$$Fc \cap Fv = [0 \quad 0 \quad 0 \quad 1 \quad 1] = 2$$

$$Fc \cup Fv = [1 \quad 1 \quad 1 \quad 0 \quad 1] = 4$$

$$S_{ff}(Pc, Pv) = 2/4 = 0.5$$

Hence according to Jaccard similarity the similarity of user 1 and user 2 can be described as 0.5 which is a moderate value. The method of calculating the threshold will be discussed in next section and until that if the assumed threshold is 0.8, then for these two users there is no possibility of becoming as a victim and clone pair.

5.7.2. Recommended friend network

A successful adversary will add the users to his network before victims adds them to his network. These friends are the people that victim knows in real time but still not friends in the OSN network. Since they are real life friends, they will most probably appear in recommended friend list (friend suggestions) of the victim. When the adversary adds them in victims clone profile, it will become more realistic and will be difficult for the victim to add them to his true profile. Under this situation, the victim's recommended friend network and the clone's friend network will appear more similarly. Hence the following Jaccard similarity formula can be used to measure the similarity between the friend list (F) of clone profile and the recommended friend list (RF) of the victim profile.

$$S_{rf}(Pc, Pv) = \frac{Fc \cap RFv}{Fc \cup RFv}$$

Equation 5.3: Friend Network Similarity between Friend list and recommended friend list of two profiles

S_{rf} – Similarity between friend list of clone and recommended friend list of victim

Fc – Friend List of Clone Profile

RFv – Recommended Friend List of Victim Profile

$Fc \cap RFv$ – Common friends between two networks

$Fc \cup Fv$ – Total friends available in two networks

5.7.3. Aggregate Friend Network Similarity

Overall friend network similarity can be calculated by aggregating the network similarities between friend networks of victim and clone (S_{ff}), recommended friend list of victim and friend list of clones (S_{rf}). The importance of those three network similarities are different [4] where the importance of S_{ff} is the higher than S_{rf} of the overall aggregate network similarity (S_n). Thus, $\boldsymbol{\alpha > \beta}$ and the literature has estimated their values based on the distribution of the data set [13][4] while some have tested the weighted values based on experimental results [10]. Thus in the present study Thus, $\alpha > \beta$ and they were calculated as $\alpha=0.9$ and $\beta=0.1$ by taking the average S_{ff} and S_{rf} between all the known clone victim pairs. After considering the weighted network similarity values, the final aggregated friend network similarity value will be calculated and forward to find the overall profile similarity of two given profiles.

$$\boldsymbol{S_n \quad (Pc, Pv) \quad = \quad (\alpha S_{ff} \quad + \quad \beta S_{rf})}$$

Equation 5.4: Aggregate Network Similarity between two profiles

5.8. Profile Similarity Threshold Calculation and Clone Detection

This research study has assumed that all other friends of both clone and victim are authentic. Considering fake profiles other than the clone in the network will be considered as future improvements of this research work.

After the calculation of aggregated network similarity between the victim and the candidate list of profiles, it will be compared with a predefined threshold value called "μ". All the candidate profiles which give a network similarity value higher than this threshold will be considered as possible clone list. In[10] the researchers have estimated this value by experimental results taken from the known list of fakes and victim sample. They have considered the change of true positive, false positive percentages etc. While another research [4] found this threshold by assigning the minimum network similarities found from the known pair of clone and victim to the Equation 6 above.

In the present research, the threshold similarity value "μ" was calculated by considering the similarity values between the known clones and victim pairs. This methodology was similar to the methodology of calculating the weights for each attribute in a previous section. However, instead of finding the average similarities attribute wise, here a total similarity value was calculated for each victim clone pairs. Finally, the minimum of them was taken as the threshold since the average value can lose some of the actual clone profiles without being detected by the threshold. An example was given in below table 5.3.

User	Attribute1	Similarity	Attribute2	similarity	Att3	Similarity	Total Similarity
Victim1	315	0	763	1	45	1	2
Clone1	2103		763		45		
Victim2	410	0	103	1	12	1	2
Clone2	NULL		103		12		
Victim3	26	1	56	0	75	0	1
Clone3	26		89		321		
	Threshold / Min						1

Table5.3: Similarity Threshold Calculation

According to the calculated profile similarity threshold value, all the profiles which has a higher similarity than this were selected as the possible clone of the given victim.

5.9. Result Validation

A new testing profile set has used to validate the process to measure the accuracy of the proposed method. If the selected profile set are actually the clones, then the accuracy of the proposed method will be high. This validation process was conducted by different researches in several ways and some of them were difficult to implement practically. In paper [23], they have taken the manual assistant from security specialists to verify the selected clone profiles. While some [29] have developed third

party Facebook apps to confirm the identity by requesting real life Identity information such as national identity card, driving license, birth certificate etc. However, in this research the validation of the result will be done using the known clone profile set and by measuring the cluster distance performances. The percentage of the accuracy will be determined by the selection of actual clone profiles as the possible profiles in the work.

Chapter 6

Implementation of the clone profile detection model

6.1. Introduction

In the previous chapters the approach and model designing techniques were discussed and, in this chapter, will present the actual implementation of model with algorithms, techniques and methodologies.

6.2. Data Preprocessing

As the first step of the implementation phase, the data refining and changing to the required formats have done. Due to the considerably large dataset this was one of the most difficult tasks in this research.

6.2.1. Friend Network dataset

In the downloaded dataset two file sets were available. First one represented the user – friend connections in the network. It was a two column 88234 rows dataset which represent all the friend connections available in the network. See figure 6.1.

However, there were only single directional connections available on the file, which means the connection not available when the 1 is the user and 0 is his friend. Hence finding the full list of friend connections for each user must be done, and it was done using Mysql. Now the new dataset became a file of 176,468 numbers of records.

user	friend
0	1
0	2
0	3
0	4
0	5
0	6
0	7
0	8
0	9
0	10
0	11
0	12
0	13
0	14
0	15
0	16
0	17

Figure 6.1: Subset of the list of friends for each user in OSN

Due to the processing difficulty of a file with very large number of records and easiness of finding the similarities between two user pairs, the above file has stored in a 4038 by 4038 adjacency matrix in MySQL. See figure 6.2

user	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
10	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 6.2: Subset of the list of friends for each user in OSN

6.2.2. Attribute feature dataset

The rest of the files of the dataset represented the feature attribute lists, 1-0 vectors representing features of individual users. However, the feature lists and 1-0 vectors were given by dividing into different circle groups. The dataset owners [24] introduce these circles as social groups such as university friends, relatives, school friends etc. and a particular user may belongs to several circles. In different circles, same user may have different features since the data owners represent the availability of only the interest features for that circle. Hence, a considerable amount of effort has taken on data extractions to get the required data into required formats. Figure 6.3 and 6.4 shows the original file and figure 6.5 show the file after transformation as a 4039 by 11 matric. The NULL values represent the private attributes of users which are not visible to the public. Hence filling of missing values were not necessary since it automatically became NULL.

Figure 6.3: Part of the Binary Vectors corresponding to the availability of features for a particular user

```
0 birthday;anonymized feature 0
1 birthday;anonymized feature 1
2 birthday;anonymized feature 2
3 birthday;anonymized feature 3
4 birthday;anonymized feature 4
5 birthday;anonymized feature 5
6 birthday;anonymized feature 6
7 birthday;anonymized feature 7
8 education;classes;id;anonymized feature 8
9 education;classes;id;anonymized feature 9
10 education;classes;id;anonymized feature 10
11 education;classes;id;anonymized feature 11
12 education;classes;id;anonymized feature 12
13 education;concentration;id;anonymized feature 13
14 education;concentration;id;anonymized feature 14
15 education;concentration;id;anonymized feature 15
16 education;concentration;id;anonymized feature 16
17 education;concentration;id;anonymized feature 17
18 education;concentration;id;anonymized feature 18
19 education;concentration;id;anonymized feature 19
20 education;degree;id;anonymized feature 20
```

Figure 6.4: Part of the anonymized Feature list corresponding to a particular user in a particular circle

user	birthday	education_school	first_name	last_name	gender	hometown	location	work_employer	work_location	work_position
0	1389	2072	2966	2479	78	173	1681	NULL	935	NULL
1	39	30	2873	2461	77	1632	567	248	1150	NULL
2	729	35	3005	114	78	1777	135	NULL	NULL	NULL
3	7	2145	557	NULL	78	1820	137	1258	137	4977
4	1317	2139	894	2793	78	132	NULL	NULL	1606	NULL
5	NULL	2073	3590	2168	78	936	1099	1933	NULL	691
6	1	NULL	557	111	78	1833	1098	636	1718	4963
7	1381	2049	557	107	78	1153	137	144	NULL	188
8	378	NULL	3597	2193	78	1151	NULL	1255	NULL	1169
9	1323	NULL	2913	2623	78	1788	NULL	292	1696	692
10	6	2140	2998	2550	78	1840	137	50	565	5007
11	NULL	NULL	3125	2186	77	1680	1147	343	NULL	NULL
12	NULL	39	1064	2334	77	1755	1152	NULL	940	NULL
13	7	2150	557	364	78	263	137	NULL	NULL	4992
14	NULL	39	3848	108	78	1640	568	NULL	129	NULL
15	NULL	30	2873	2461	77	1632	NULL	NULL	1150	NULL
16	1502	NULL	3333	2173	78	1630	176	NULL	NULL	699

Figure 6.5: Attribute adjacency matrix (4039*11)

6.3. Recommended friend list generation

First the non-friend list for all the users has generated. Then, by only considering the school, hometown, location and work employer the similarity between each two-

profile pair was calculated. Finally, the mutual friend similarity between each user pair was calculated. For these attribute and friend similarity calculations, the Jaccard similarity measurement was used.

Finally, by filtering the non-friends who have higher mutual friend similarity along with a higher attribute similarity were taken as the list of recommended friends for each user. The generated number of recommended friends for a particular user was around 20-30.

6.4. The Model

After preprocessing the data, the detection phase has started. The model development was done using RapidMiner 9.1 tool hence the required data files, friend adjacency matrix, attribute adjacency matrix, recommended adjacency matrix were loaded into the repository and used for the two processes created for each detection phase 2 and phase 3.

6.5. Detection Phase 1 – Filter by Names

The first stage of the detection process contains the filter users by their first_names. The model diagram for phase 1 is given in figure 6.6.

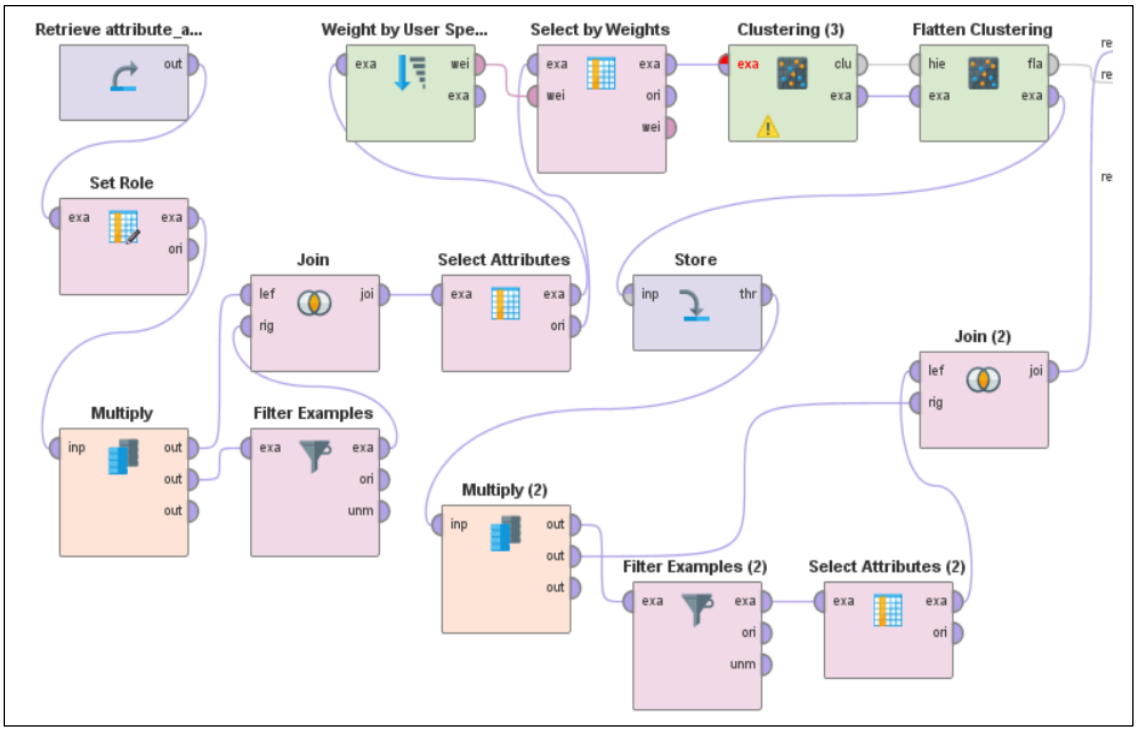

Figure 6.6: Detection phase 1 and 2- system model

First the attribute adjacency matrix data set was called using the retrieve operator. All the data columns have changed into polynomial (categorical type with many possible values). Then the set role operator has used to assign the id role to the user column since it is necessary before passing to the next operator named the Filter Example. Using the custom filter option in Filter Example operator the victim ID was given to the model. See figure 6.7

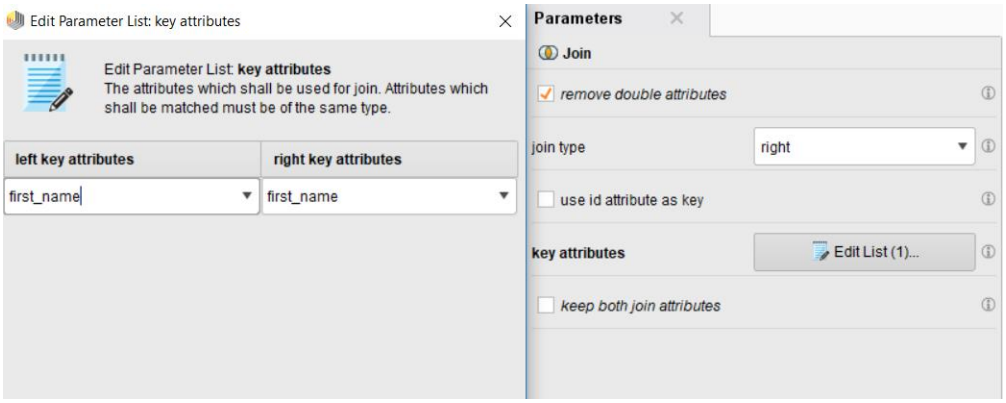

Figure 6.7 : Join by frist_names

And the filtered victim record was then joined with the original adjacency matrix based on the first_name and from those records all the attributes except the first_name was selected using the inversion of Select Attribute operator.

6.6. Detection Phase 2 – Clustering on categorical data

In the next step each attributes of the filtered data set were assigned a pre-estimated weight value as given in the figure 6.8 and select the all the attributes for the clustering step by take top p% as 1.0 in “Select By Weights” operator.

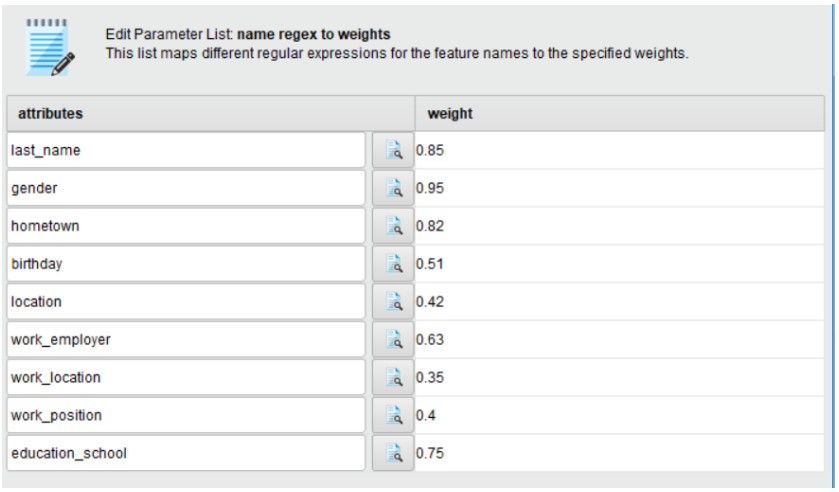

Figure 6.8: weight values estimated for each attribute

Then based on these filtered and weighted user attribute set, the candidate profiles were grouped into clusters by applying Agglomerative algorithm using nominal measurement and nominal distance. The number of clusters was set to 6, according to the findings in previous chapter. See figure 6.9 for clustering specifications

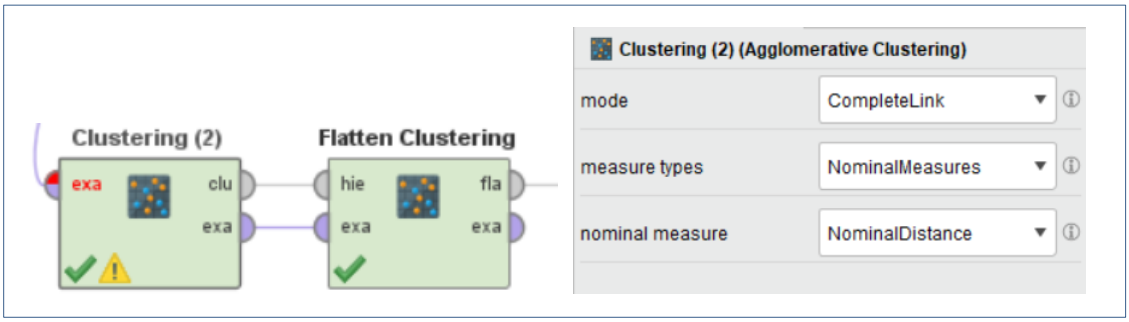

Figure 6.9 Clustering criteria in RapidMiner

In the next step the cluster object was stored into the repository and one of it has sent to the Filter Example followed by the Select Attribute operator to filter only the cluster members belongs to the same cluster of the victim. Finally, a joined was performed to get all the attributes and the cluster information as a single output. Figure 6.10 shows the final outcomes of detection phase 2.

Row No.	user	cluster	first_name	birthday	education_school	last_name	gender	hometown	location	work_employer	work_location	work_position
1	48	cluster_0	3701	735	2130	2187	77	1842	132	NULL	NULL	4994
2	1381	cluster_0	3701	5	237	590	77	1148	NULL	1914	NULL	4998
3	1926	cluster_0	3701	NULL	444	2342	77	908	176	970	1740	4970
4	2079	cluster_0	3701	735	NULL	2187	77	1842	132	NULL	NULL	4994
5	3500	cluster_0	3701	1305	1197	106	77	1685	1706	1902	NULL	4975

Cluster Model

```
Cluster 0: 5 items
Cluster 1: 10 items
Cluster 2: 29 items
Total number of items: 44
```

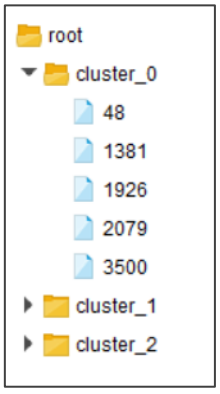

Figure 6.10: clustering output for a given victim

6.7. Detection Phase 3 – Friend Network Similarity Calculation

The final stage of the detection process was finding the percentage of the possibility of becoming a clone using a calculated network similarity measure. The output from the detection phase 3 has used to further verify the duplicability of a profile. The model developed here is given in figure 6.11 below.

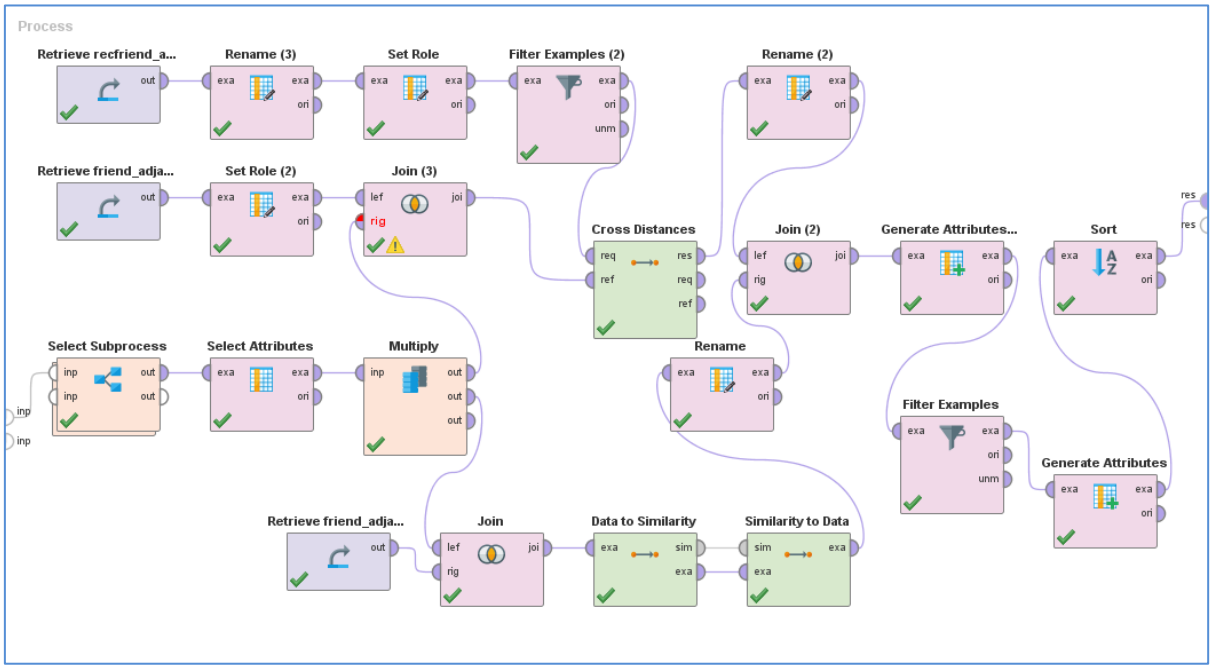

Figure6.11: detection phase 3 - network similarity

The previous process has saved a sub process to the new phase. Previous output was combined with the friend and recommended friend adjacency matrices and selects only the friends columns. The Data to Similarity and Similarity to Data operators have used to calculate the similarity based on friend networks and recommended friend networks based on Nominal measures, Jaccard similarity. Then the Filter Example operator has used to filter the similarity by the given threshold. Next using the Generate Attributes operator calculates the percentage of possibility of labeling as a clone according to the equation 6.1 given below.

Clone Percentage = (SIMILARITY – 0.93) / (1 – 0.93) * 100

Equation 6.1: Clone Percentage Calculation

The output from the detection phase 3 was given in the below figure 6.13

Row No.	FIRST_ID	SECOND_ID	SIMILARITY	Clone Perce...
1	48	2079	0.994	97.153
2	48	3500	0.990	94.801
3	48	1381	0.988	93.934
4	48	1926	0.961	80.688

Figure 6.12: Calculated similarity and percentage

6.8. Calculating the similarity threshold value

The minimum threshold taken by the known pairs was 0.93 and all the suspect profiles having a similarity value higher than this value were given a percentage indicating that how much similar they are to the genuine victim. The application should have a real time validation mechanism to verify the actual clone profiles.

6.9. Testing Phase

After building and training the detection model using an unsupervised learning method and statistical similarity measurement, the model was ready to be tested on an unknown dataset. Since it was unable to divide the dataset into a testing set due to the higher dependencies between each user based on friend networks, the model was tested by a new dataset contains with 2000 user profiles generated artificially.

The artificial test data set was generated using a data generator tool called “Advanced Data Generator” and it easily generated data into MySQL tables for further processing.

Chapter 7

Results Evaluation & Discussion on Work

7.1. Introduction

The previous chapter presented the implementation of the model and the phases of it. The results gain from the implementation phase will be interpreted here in order to make decisions in the final chapter. The evaluation of the results of clustering and other statistical data will be corresponds to the testing results of the model.

7.2. Clustering as a supervise method

Clustering can be done as a fully unsupervised learning mechanism while some time clustering can have some known output data such as the prior knowledge of the cluster numbers in which an object will fall in. This type of clustering is called the "Gold Standard" and this will help to conduct a comprehensive comparison of the accuracies of different clustering methods. Of course, up to some extend this research is also analyzing the answer using the prior knowledge of objects and clusters, however the number of label information is comparatively less since only clones are 2%of the dataset. Otherwise the Map Clustering on Labels operator from the RapidMiner can be used with performance operator to evaluate the measurement. Here the labeled has used to make sure the algorithm tries to find those clusters correctly and to calculate some statistical values needed by the rest of the algorithm such as similarity threshold and attribute weights.

Example 1

- Cluster for the first_name : 3701
- Number of users : 44
  - Cluster 0 : 5
  - Cluster 1 : 10
  - Cluster 2 : 29

Attribute	cluster_0	cluster_1	cluster_2
birthday	87	24	51.034
education_school	150.800	123	60.690
last_name	267.600	359.700	335.517
gender	2	1	1.448
hometown	169.600	181.300	127.103
location	53.400	58.400	79.552
work_employer	85.400	104.700	1
work_location	28	30.300	42.621
work_position	93.200	51.900	1

Figure7.1: Cluster centroid table 1

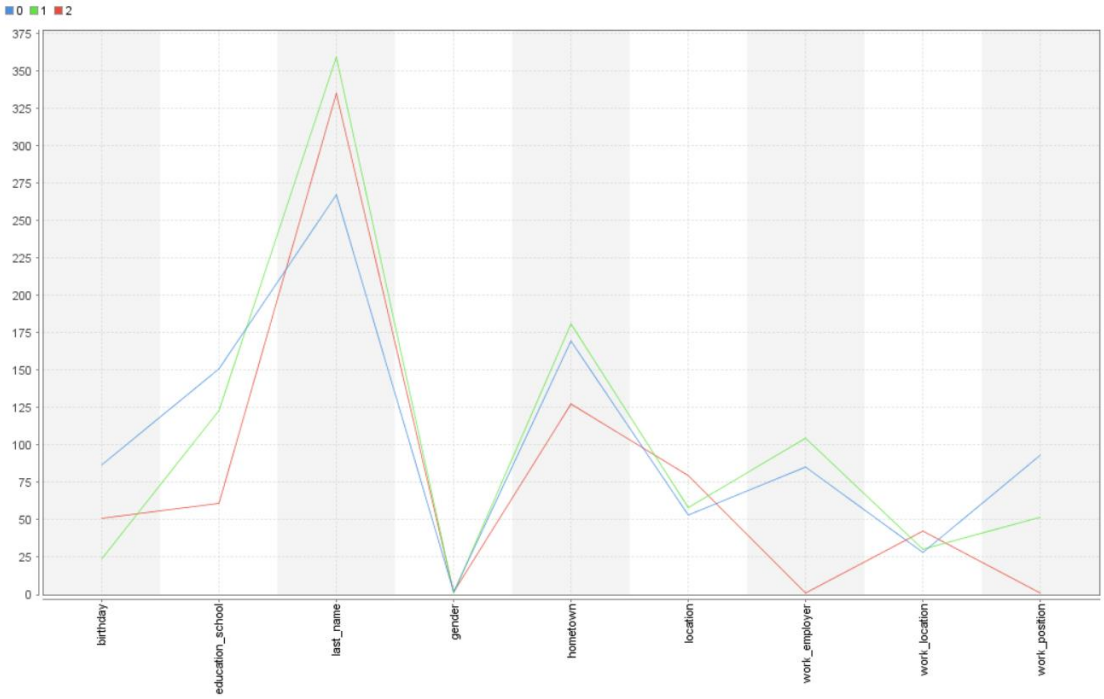

Figure 7.2: Cluster allocation plot 1

According to the centroid table and the plot more details about the three clusters can be derived. Last_name seems to be played a major role in all the three clusters where prominent in cluster 1.

Example 2

- Cluster for the first_name : 3005
- Number of users : 37
  - Cluster 0 : 13
  - Cluster 1 : 6
  - Cluster 2 : 18

Attribute	cluster_0	cluster_1	cluster_2
birthday	87	24	51.034
education_school	150.800	123	60.690
last_name	267.600	359.700	335.517
gender	2	1	1.448
hometown	169.600	181.300	127.103
location	53.400	58.400	79.552
work_employer	85.400	104.700	1
work_location	28	30.300	42.621
work_position	93.200	51.900	1

Figure 7.3: Cluster centroid table 2

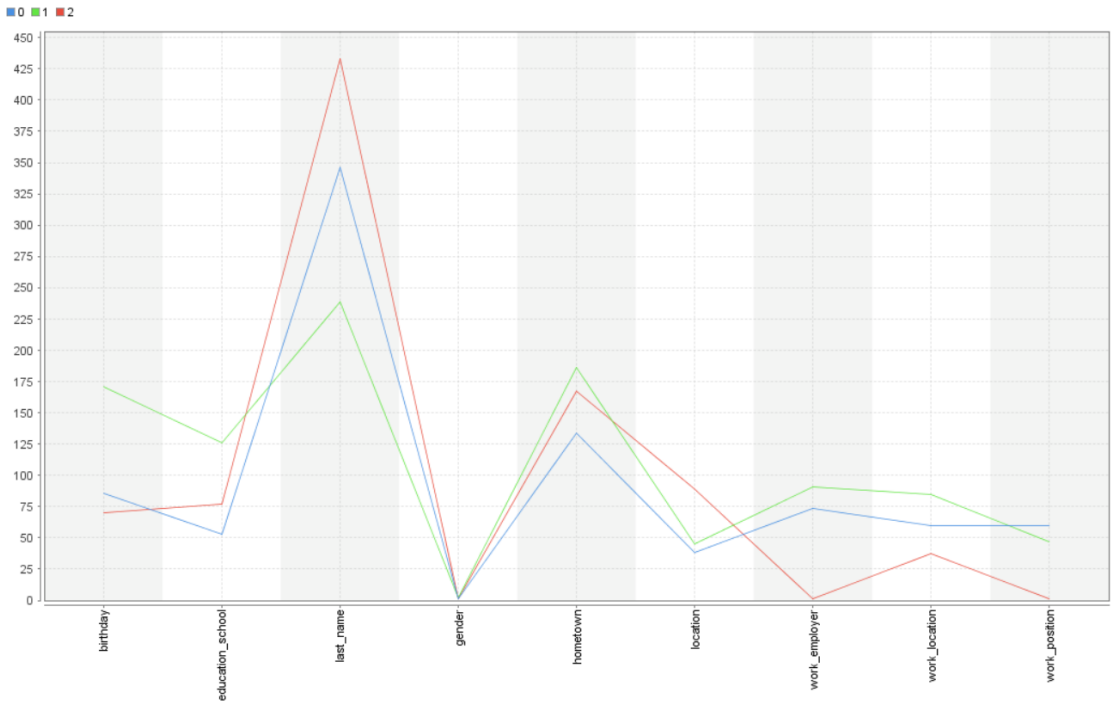

Figure7.4: Cluster allocation plot 2

Example 3

- Cluster for the first_name : 3125
- Number of users : 49
  - Cluster 0 : 9
  - Cluster 1 : 27
  - Cluster 2 : 13

Attribute	cluster_0	cluster_1	cluster_2
birthday	84.111	82.852	13.846
education_school	72.444	69.296	44.846
last_name	343.556	359.333	266.692
gender	2	1	2
hometown	112.889	169.778	103.231
location	53.778	74.259	37.923
work_employer	19.889	58.889	53.615
work_location	60.333	34.074	3
work_position	21.222	28.815	20

Figure 7.5: Cluster centroid table 3

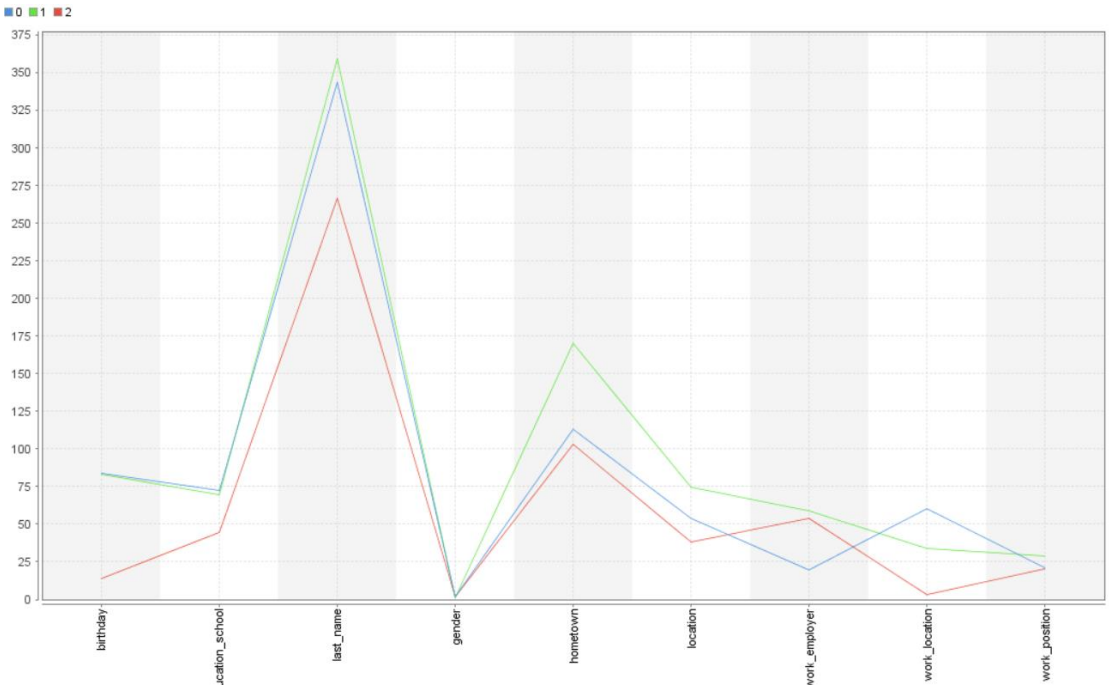

Figure 7.6: Cluster allocation plot 3

7.3. Network Similarity performance evaluation

Following figures shows the resulting tables and plots related to similarity and clone percentage of few selected victims. They were taken from the above victim-clone pair example and the differences between similarity percentage of actual clone and the other predicted clones are distinguishable in many known pairs.

- **Victim ID : 48 Clone ID: 2079**

Row No.	FIRST_ID	SECOND_ID	SIMILARITY	Clone Perce...
1	48	2079	0.994	71.528
2	48	3500	0.990	48.007
3	48	1381	0.988	39.341

Figure7.7: clone similarity and percentage 1

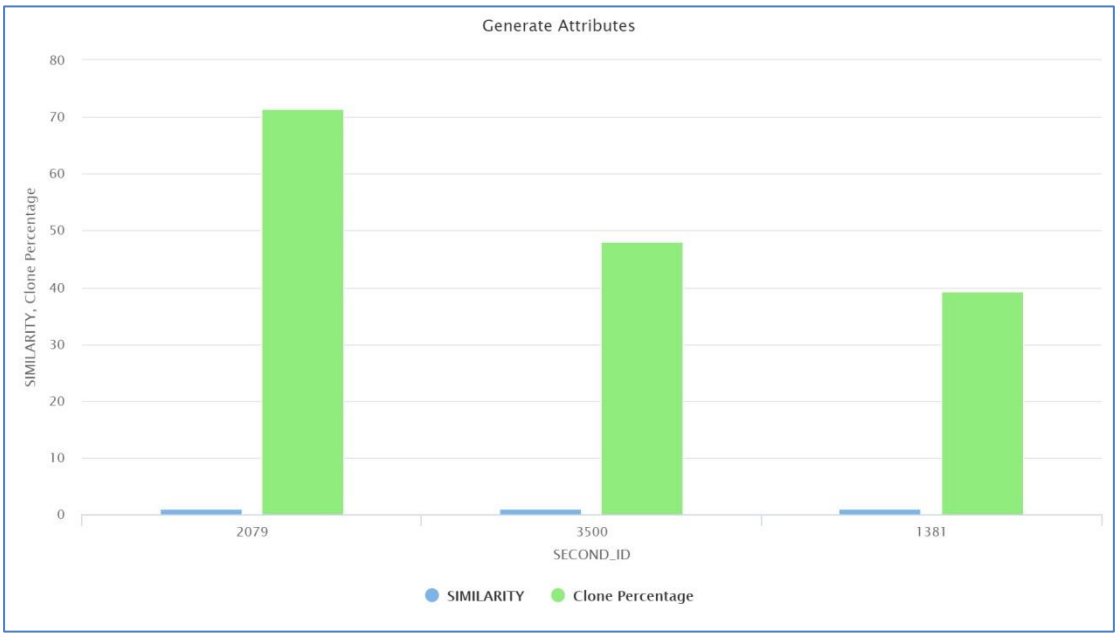

Figure 7.8: clone similarity and percentage plot 4

The actual clones was filtered as the highest possible clone with a 71.52% . Other two profiles have a significantly low percentages compared to the actual one. Hence the accuracy is high

The plot shows the results when number of clusters are 3, and there is a significant difference between the actual clones and other possible profiles.

- **Victim ID : 2 Clone ID: 1162**

Row No.	FIRST_ID	SECOND_ID	SIMILARITY	Clone Perce...
1	2	3716	0.996	78.955
2	2	3899	0.996	78.955
3	2	1103	0.996	77.717
4	2	1162	0.996	77.717
5	2	3583	0.995	75.241
6	2	2177	0.995	74.003
7	2	3784	0.995	72.766
8	2	1936	0.992	60.386
9	2	1674	0.991	56.672
10	2	814	0.991	55.435
11	2	3518	0.989	46.769
12	2	2823	0.988	40.579
13	2	853	0.987	33.152
14	2	1404	0.985	25.724

Figure 7.9: clone similarity and percentage 2

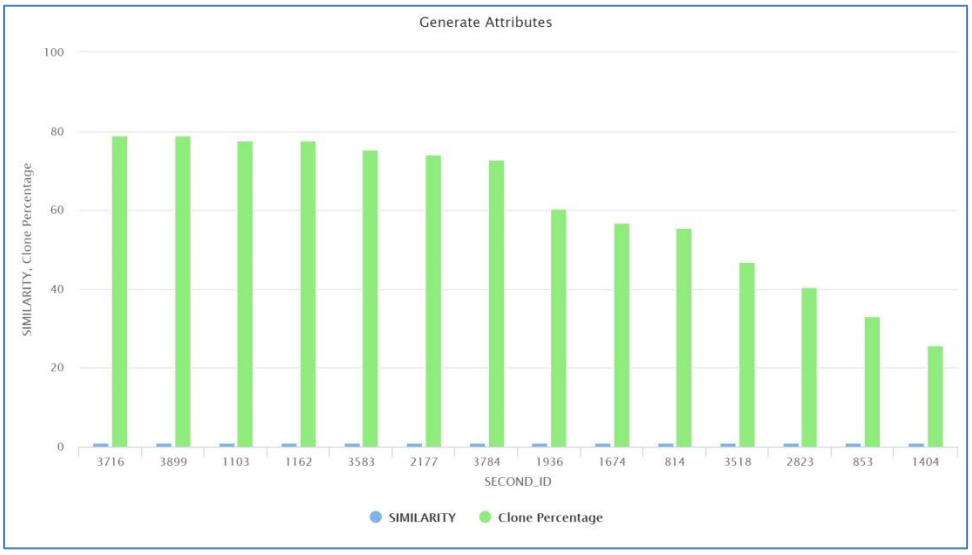

Figure 7.10: clone similarity and percentage plot 5

- **Victim ID : 11 Clone ID: 1693, 1251, 885**

Row No.	FIRST_ID	SECOND_ID	SIMILARITY	Clone Perce...
1	11	1276	0.999	96.286
2	11	1693	0.999	95.048
3	11	885	0.998	87.621
4	11	1251	0.997	83.907
5	11	2440	0.991	56.672
6	11	1787	0.990	51.721
7	11	1812	0.989	44.293
8	11	2909	0.981	3.441

Figure 7.11: clone similarity and percentage 3

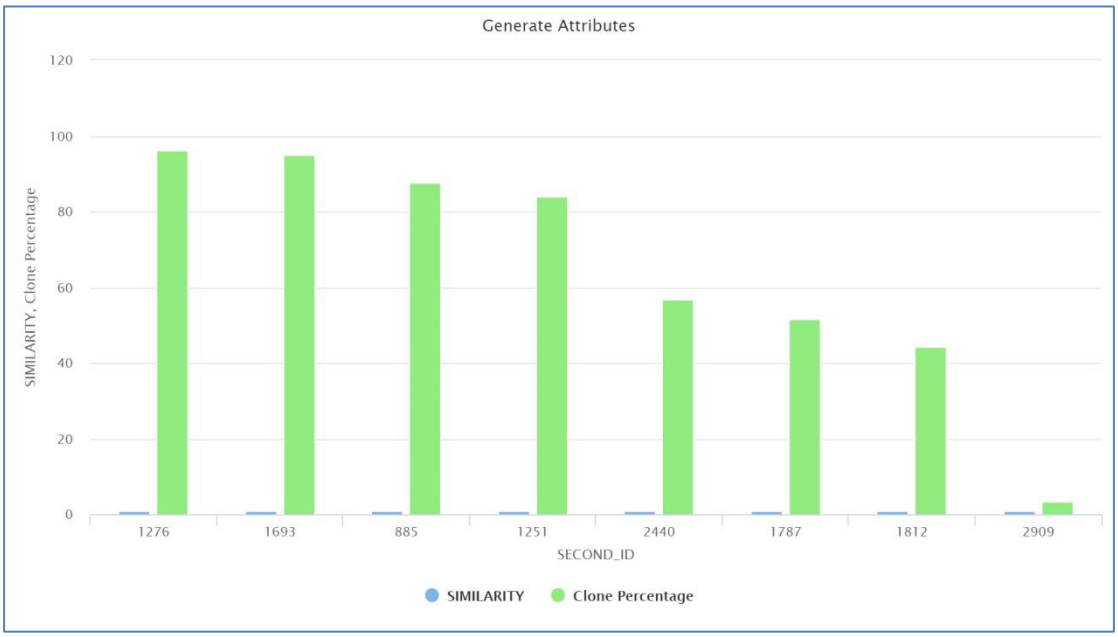

Figure 7.12: clone similarity and percentage plot 6

7.4. The Results Validation

Nevertheless, considering all the victim-clone pair examples the Precision [TP/ (TP + FP), where TP – examples selected the actual clone as the possible clones with the highest clone percentage and FP – False Positive, examples did not select the actual clone as the possible clones with the highest clone percentage] was 88.75% and it was considerably a good performance.

Moreover, the performance of the system highly depends on the clustering technique in which most of the similar profiles will be filtered out from a large sample. Hence the selection of a suitable clustering algorithm and a similarity or distance measurement is crucial. The density-based cluster performance evaluation was used to evaluate the performance of the clustering method and it gave relatively low average within cluster distance values for most of the examples where it was -50.446 for the above example. The performance evaluation model is given below in figure 7.13

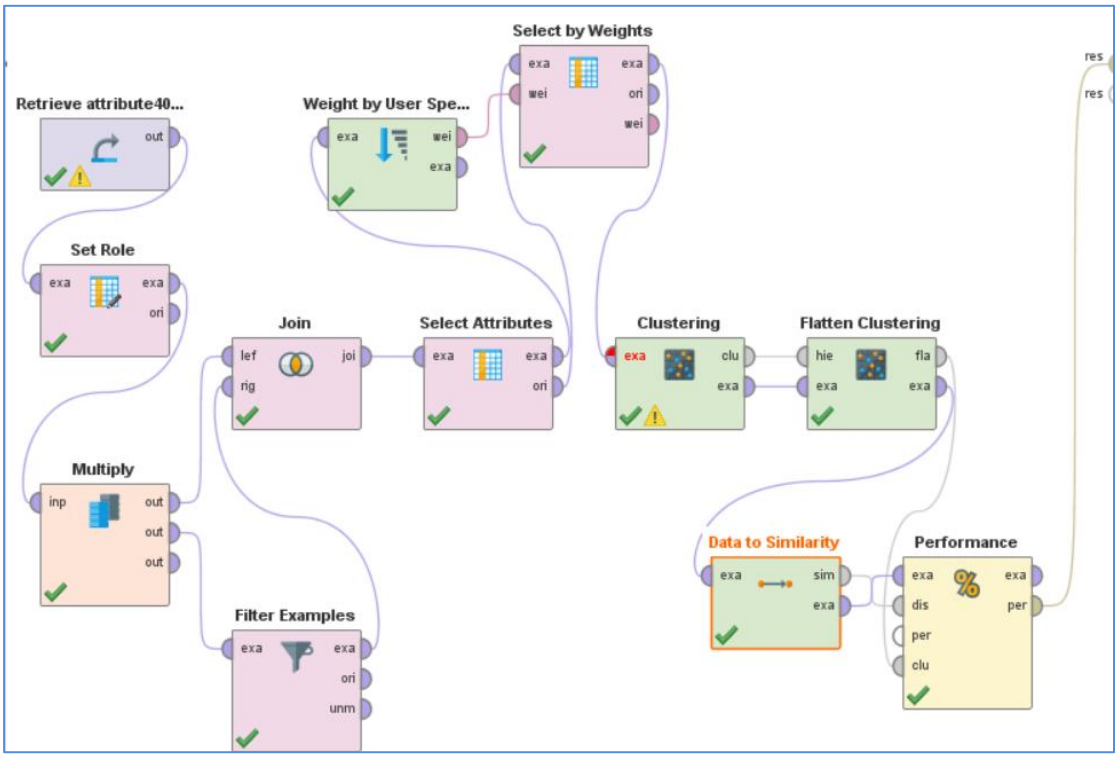

Figure 7.13: Cluster density performance

Results performance vector

Avg. within cluster distance: -50.446
Avg. within cluster distance for cluster 0: -10.217
Avg. within cluster distance for cluster 1: -24.118
Avg. within cluster distance for cluster 2: -66.461

Chapter 8

Conclusion & Future Work

8.1. Introduction

Among most of the security issues in online social networks, fake profile gets more importance since it can lead to severe security and user privacy threats. Identity Clone Attack is one of the fake profile problems which was considered as the most dangerous threat in OSN. Hence the detection of clone profiles has become an important area in the research field of computer science all over the world and 75 percent of the existing solutions were found after 2010.

The detection of clone profiles in social networks is a currently engaging research problem and most of the investigations are done using Facebook, as it is the most popular social network platform. Other than that Twitter is also a widely used network since there are less privacy concerns when creating user profiles. When considering the selected platforms of past researches, the networks having less complex process for creating user profiles and weak user authentication mechanisms have mostly been subjected to the clone profile issue.

8.2. Challenges to the field of research

Detection of clone profiles in OSN is a difficult task due to several reasons.

- **Considering 'Name' as the initially filtering feature**

Real Profiles can have almost same names but still genuine[9]. Hence doing the initial filtration by using names of the user will give incorrect result as they are clones. On the other hand due to feature anonymization, almost similar names with little changes will not be detected as similar.

Example: Though Harris Patrick Kevilton can be cloned as Harris P Kevilton, due to anonymization of features those two will be feature1 and feature2. Hence two accounts will not be detected as clones.

Also when searching profiles matches, victim profile will be compared with a large pool of profiles where the time taken can be significantly high. Hence the efficiency

can be improved, if the initial filtration can be done after considering the network circles in the Facebook network. And also by using an indexing method to search matching profiles would increase the efficiency of the algorithm.

- **Dynamic and Diverse nature of OSNs**

Social network is a highly rapidly changing diverse environment due to users with various characters and behaviors. Under these conditions predicting user behaviors and making assumptions is very difficult.

Example: A fake clone user can have many friends of the victim and actively participate on commenting and like events. Meanwhile some fake users just want to be pretend as another user just to stay on the network with a hidden true identity. He might not involve in any active event but still fake.

- **Network with multiple fake identities**

When there is more than one clone profile in the considered network (after the initial filtration of profiles by name) it will be difficult to get the correct values for the similarity measurements since their attributes or friend networks are not genuine. In the proposed method Assume only the clone of the victim is the fake profile and further improvements to the method by considering multiple clone profiles can be done under future improvements.

8.3. Challenges to the proposed model

- **Finding a featured data set**

Some researchers [9] have used synthetic data sets for their investigations and these may not give the most realistic solutions since social networks are highly diverse environments and this complex diversity can be efficiently gain using real data. However still most of the researches made their assumptions using very limited amount of real data since it is difficult to get personal user data through an API due to confined accessibilities. Most of the researches doing their work on the Facebook platform are facing this problem. Hence finding a better dataset with lots of interesting attributes was the main challenge of the research.

- **Inability of verifying the results in a real time system**

Besides the featured dataset, finding a verified dataset of clone profile was also very difficult since Facebook is highly considering their internal privacy. Thus, same as many past researchers, here also a synthetic clone profile set will be created under lots of assumptions. Under this circumstance the validation of the results in a real time system will be unable.

- **Hardware Resource limitations**

Due to large amount of data set the existing hardware facilities were not enough to perform the research work efficiently. To find the friend network similarity between 4039 users took more than 9 hours in a core i7, 8GB laptop.

8.4. How this study can be modified in future

- The model can be tested with more than 10 attributes to identify the relationships between different attributes and clone profiles.
- Use dimensionality reduction similarity algorithms such as minHash, SimHash rather than using direct similarity measurement like Jaccard, Cosine etc.
- Use of more sophisticated cluster evaluation methods such as, F measure, Rand Index, Purity.
- String matching functions can be used to match the actual text of a name when non-anonymized features are given
- In addition, the main future interest of this research study is to build a model to detect the actual person behind this clone who created the clone profile by analyzing the behavioral patterns of profiles in OSN.

8.5. Conclusion

The threat of creating clone profiles has been increased over the past years with the popularity and the simplicity of making profiles on Facebook. With this attack the personal information of users can be misused and can cause damages to their good reputation. Though there were different methodologies for detecting clone profiles in OSNs, due to the diverse characteristics and rapidly changing nature of social networks, faked clone profile detection was still not fully solved by existing approaches and opened for future directions. This paper introduces a model with three main stages to detect these clone profiles on Facebook where in each stage the amount of computation to be done was reduced by filtering profiles in each of the stages. This was a simple but more effective method that also showed a higher precision. Furthermore, since most of the calculations are done considering the distribution of the dataset, this model can be easily adjusted to a different dataset by only finding values for few parameters.

References

[1] Statista, “Social Media Statistics & Facts,” 2017. [Online]. Available: https://www.statista.com/topics/1164/social-networks/. [Accessed: 30-Oct-2017].

[2] WordStream, “40 Essential Social Media Marketing Statistics for 2017,” 2017. [Online]. Available: http://www.wordstream.com/blog/ws/2017/01/05/social-media-marketing-statistics. [Accessed: 10-Nov-2017].

[3] F. Rizi, M. Khayyambashi, and M. Kharaji, “A New Approach for Finding Cloned Profiles in Online Social Networks,” *Int. J. Netw. Secur.*, vol. 6, no. April, pp. 25–37, 2014.

[4] L. Jin, H. Takabi, and J. B. D. Joshi, “Towards active detection of identity clone attacks on online social networks,” *Proc. first ACM Conf. Data Appl. Secur. Priv. - CODASPY ’11*, p. 27, 2011.

[5] N. Kumar and R. N. Reddy, “Automatic Detection of Fake Profiles in Online Social Networks,” National Institute of Technology Rourkela Rourkela-769 008, Orissa, India, 2012.

[6] P. Dewan, S. Bagroy, and P. Kumaraguru, “Hiding in Plain Sight : Characterizing and Detecting Malicious Facebook Pages,” pp. 193–196, 2016.

[7] M. Torky, A. Meligy, and H. Ibrahim, “Recognizing fake identities in online social networks based on a finite automaton approach,” *2016 12th Int. Comput. Eng. Conf. ICENCO 2016 Boundless Smart Soc.*, pp. 1–7, 2017.

[8] K. Krombholz, D. Merkl, and E. Weippl, “Fake identities in social media: A case study on the sustainability of the Facebook business model,” *J. Serv. Sci. Res.*, vol. 4, no. 2, pp. 175–212, 2012.

[9] G. A. Kamhoua *et al.*, “Preventing Colluding Identity Clone Attacks in Online Social Networks,” in *2017 IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW)*, 2017, pp. 187–192.

[10] P. Bródka, M. Sobas, and H. Johnson, “Profile cloning detection in social networks,” *Proc. - 2014 Eur. Netw. Intell. Conf. ENIC 2014*, pp. 63–68, 2014.

[11] M. A. Devmane and N. K. Rana, “Detection and prevention of profile cloning in online social networks,” *Int. Conf. Recent Adv. Innov. Eng. ICRAIE 2014*, pp. 9–13, 2014.

[12] G. Kontaxis, I. Polakis, S. Ioannidis, and E. P. Markatos, “Detecting social network profile cloning,” *2011 IEEE Int. Conf. Pervasive Comput. Commun. Work. PERCOM Work. 2011*, pp. 295–300, 2011.

[13] M. R. Khayyambashi and F. S. Rizi, “An approach for detecting profile cloning in online social networks,” *2013 7th Intenational Conf. e-Commerce Dev. Ctries. With Focus e-Security, ECDC 2013*, pp. 1–12, 2013.

[14] M. A. Wani and S. Jabin, “A sneak into the Devil’s Colony - Fake Profiles in

Online Social Networks," 2017.

[15] B. B. Das, "Profile Similarity Technique for Detection of Duplicate Profiles in Online Social Network," vol. 7, no. 2, pp. 507–512, 2016.

[16] M. Egele, C. Kruegel, and G. Vigna, "C OMPA : Detecting Compromised Accounts on Social Networks."

[17] A. Malhotra, L. Totti, W. Meira, P. Kumaraguru, and V. Almeida, "Studying user footprints in different online social networks," *Proc. 2012 IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Mining, ASONAM 2012*, pp. 1065–1070, 2013.

[18] A. M. Meligy, "A Framework for Detecting Cloning Attacks in OSN Based on a Novel Social Graph Topology," no. February, pp. 13–20, 2015.

[19] M. Zabielski, R. Kasprzyk, Z. Tarapata, and K. Szkółka, "Methods of Profile Cloning Detection in Online Social Networks," *MATEC Web Conf.*, vol. 76, 2016.

[20] F. S. Rizi and M. R. Khayyambashi, "Profile Cloning in Online Social Networks," *Int. J. Comput. Sci. Inf. Secur.*, vol. 11, no. 8, pp. 82–86, 2013.

[21] H. Agrawal and R. Kaushal, "Analysis of Text Mining Techniques over Public Pages of Facebook," in *Proceedings - 6th International Advanced Computing Conference, IACC 2016*, 2016, pp. 9–14.

[22] M. Kharaji and F. Rizi, "An IAC Approach for Detecting Profile Cloning in Online Social Networks," *Int. J. Netw. Secur. Its Appl.*, vol. 6, no. 1, pp. 75–90, 2014.

[23] Q. Cao, X. Yang, J. Yu, and C. Palow, "Uncovering Large Groups of Active Malicious Accounts in Online Social Networks," *Proc. 2014 ACM SIGSAC Conf. Comput. Commun. Secur. - CCS '14*, pp. 477–488, 2014.

[24] J. Lescovec, "Stanford University - Data Repository," 2012. [Online]. Available: https://snap.stanford.edu/data/egonets-Facebook.html. [Accessed: 10-May-2018].

[25] S. Mazhari, S. M. Fakhrahmad, and H. Sadeghbeygi, "A user-profile-based friendship recommendation solution in social networks," *J. Inf. Sci.*, vol. 41, no. 3, pp. 284–295, 2015.

[26] D. Dave, N. Mishra, and S. Sharma, "Detection Techniques of Clone Attack on Online Social Networks : Survey and Analysis," pp. 179–186.

[27] "Similarity Measurements," 2017. .

[28] Facebook, "Finding Friends and people you may know," 2018. [Online]. Available: www.facebook.com/help/www/336320879782850. [Accessed: 10-Dec-2018].

[29] Facebook, "Confirm Your Identity with Facebook," 2018. .

[27] Facebook, "Finding Friends and people you may know " 2018. [Online].

Available: https://www.facebook.com/help/www/336320879782850.

[28] S. Mazhari, S. M. Fakhrahmad, and H. Sadeghbeygi, “A User Profile-based friendship recommendation solution in social networks,” *Proc. 2015JIS SAGE Journal of Information Science.. -’41*, pp. 284–295, 2015