University of BRISTOL

DEPARTMENT OF COMPUTER SCIENCE

Identifying Hackers' Roles and Examining the Evolution Pattern of Hackers on Hacker Forums by Clustering and Semi-supervised Learning Algorithms

Taiquan Yuan

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of Master of Science in the Faculty of Engineering

September, 2020 | CSMSC-20

Declaration

I declare that the work in this dissertation was carried out in accordance with the requirements of the University’s Regulations and Code of Practice for Taught Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, this work is my own work. Work done in collaboration with, or with the assistance of others, is indicated as such. I have identified all material in this dissertation which is not my own work through appropriate referencing and acknowledgement. Where I have quoted or otherwise incorporated material which is the work of others, I have included the source in the references. Any views expressed in the dissertation, other than referenced material, are those of the author.

Taiquan Yuan, September 2020

Contents

Executive Summary i
Acknowledgement ii
1 Introduction 1
1.1 Topic Involved in This Research 1
1.2 Aims and Objectives 1
1.3 Structure of Dissertation 2
2 Background and Related work 3
2.1 Hacker Forum 3
2.2 Hacker Identification 5
2.3 Temporal Analysis on Hacker Community 6
2.4 Chapter Conclusion 7
3 Theoretical Background 8
3.1 Machine Learning 8
3.1.1 Supervised Learning 10
3.1.2 Unsupervised Learning 10
3.1.3 Semi-supervised Learning 11
3.2 Text Representation for Machine Learning 12
3.2.1 TF-IDF 12
3.2.2 The Mean of Word2vec 13
4 Methodology 14
4.1 Dataset 15
4.1.1 Data Source 15
4.1.2 Data Pre-processing 15
4.2 Features Extraction 16
4.2.1 Statistical Features 16
4.2.2 Latent Features 19
4.3 Clustering 22
4.4 Temporal Analysis 23
5 Results 24
5.1 Text Representation and Semi-supervised Learning 24
5.2 Features Exploration 25
5.2.1 Statistical Features 25
5.2.2 Latent Features 26
5.3 Clustering Results 27
5.4 Temporal Analysis 34
5.4.1 Between Five Groups 34
5.4.2 Within groups 35
6 Discussion and Limitations 37
6.1 Discussion 37
6.2 Limitations 38
7 Conclusion and Future Work 39

Bibliography.. 40

Executive Summary

The underground hacker forums serve as the entrance into cybercrime, where attack tools and services can be obtained at a low cost or even for free. Many people are brought together on hacker forums for different purposes such as learning hacking skills, advertising new attack tools and earning reputation by provide more detailed replies, etc. This research sought to identify these users' roles according to their purposes or activities and examine their evolution pattern to locate potential threads. Several steps of data science approaches were implemented to achieve these two goals and a well known hacker forum was chosen to do the test.

Two-step of machine learning algorithms were applied to identify users' roles. The result yielded 5 domains: producers, providers, teachers, learners and others. Each of them have several sub categories which represent different levels of users within the domains. The qualitative analysis on the 5 domains offered meaningful interpretations into the nature of the forum user base. Temporal analysis was also designed to track the transitions of roles of hackers on the forum. Users whose participation gradually decreased were captured along with learners who grew from a low-level hacker to a high skilled one.

The main contributions are summarized below:

1. A practical and effective method to identify users' roles on hacker forums was designed which can also be used in other platforms. Some experiments were also included to compare the performance of different models.
2. A qualitative analysis was conducted to reveal more details hidden in hacker forums, such as the social structures of hackers, the more detailed role classifications and the engagement and technical level of different groups of users.
3. A temporal analysis was designed to examine the evolution patterns of roles of hackers, such as how many learners became teachers during their active time on the forum.

Acknowledgement

A massive thank you to Doctor Matthew Edwards for his expert guidance, understanding, patience and support in this project. He asked about my progress every week, and he always responds as soon as I need help, without which I may not finish this project on time.

1 Introduction

1.1 Topic Involved in This Research

Global e-commerce has experienced continued growth and become more streamlined and accessible which benefits from the development of information technology. However, technology is a double-edged sword. Some people use technology to make people's life more convenient, while others use technology vulnerabilities to attack servers or spread network viruses for profit or to enjoy the challenge of their activities. Such criminality is to a great extent promoted by underground economy where crimeware and illegal assets are traded (Allodi, 2017).

The underground shadow economy can be divided into four layers based on visibility (Lusthaus, 2019), where online hacking forums are at the topmost layer. Hacker forums are considered the first door to cybercrime where hacking skills and tools can be easily accessible. On hacker forums, hackers discuss technical issues by posting and replying which can be seen by all users. To earn more reputation, skilled hackers are willing to reply with detailed posts, which lowers the threshold for committing cybercrime (Zhang and Li, 2013).

In order to extract potential threats, many studies (see Chapter 2) have been carried out to identify hackers especially key hackers and theirs specialties. On the other hand, some researchers focused on identifying hackers' roles, such as teachers to provide advice or learners to seek for solutions. Hackers' trajectory and evolution pattern are also important such as how hackers' interests and activities changes over time, which helps to better understand hackers' social dynamics and predict potential cybercrime.

1.2 Aims and Objectives

Based on past studies, this essay focuses on identifying hackers' roles on hacker forums and examining the evolution patterns of hacker's roles. The are three objectives need to be satisfied in terms of this two topic:

1. Design an automatic and reliable method to identify hackers' roles.
2. Provide a detailed qualitative analysis on the result of hackers' roles.

3. Design a proper temporal analysis on hackers' roles to examine hacker forum's social dynamics.

1.3 Structure of Dissertation

This dissertation will consist of six main chapters outlined below:

1. Background and Related Work. This chapter will provide a literature review on hacker community studies related to this essay to identify research gaps whilst providing the direction for this research.
2. Theoretical Background. This chapter will provide an overview of technical background required for readers to better understand this thesis.
3. Methodology. This chapter will show the details of the overall research design from data preparation, model selection to final analysis.
4. Results. This chapter will provide details of the experiments performed according to the research design and delivers details of the result and analysis.
5. Discussion and Limitation. This chapter will compare the findings in this essay to previous research and identify limitations of this research.
6. Conclusion and Future Work. This chapter will detail and summarize the findings in this research and give suggestions for future work.

2 Background and Related work

This chapter will present a literature review on hacker community research. The literature were ether suggested by my supervisor or obtained from a freely available scientific research database by searching for keywords such as “hacker community”, ”identify hackers”, etc or acquired by following the references of the obtained results.

The existing literature related to this research can be grouped into three streams: (1) previous hacker forum research, (2) hacker identification research, and (3) temporal analysis research on hacker forums. The work done in this thesis belongs to the latter two and therefore more details are provided on these two streams.

2.1 Hacker Forum

The increase of cybercrime incidents such as using DDOS to take down gaming servers and booters to to attack medium-size websites (Karami and Mccoy, 2013) parallels the development of information technology. Underground hacker forums serve as an entrance into cybercrime world for potential criminals, where hacker tools and services are easily accessible at low cost or even for free (Allodi, 2017). Prior research have provided a very comprehensive observation on hacker forums including forum’s operation and social structure, hackers’ interactions and hacker assets, etc.

Most hacker forums consists of multiple sub-forums related to specific topics like “Developer Section”, “Source Code” and “General Discussion”. Within each sub-forum, hackers may create a new thread by making a post or reply to the existing ongoing threads. Each forum member has a public profile used to track user’s trustworthiness rate with information such as the registration rate (Kigerl, 2020), last access, time spent and total posts. Generally, two types of managers are set on hacker forums. Administrators is at the top of the pyramid and is responsible for the overall management and decision-making of the forum. Within each sub-forum, moderators act like law-executors, banning users who do not act according to the rules and setting guidelines. The reasons for banning may vary for different platforms, but duplicate accounts, spamming and malicious activities (Allodi et al., 2015) are the most common.

The size of the forum will also lead to differences in structure. Larger forums can have thousands of members with multiple tiers and cover all areas of hacking technology. Smaller communities are limited to a couple of hundreds users with flatter hierarchies and focus on smaller subset of hacking activities (Garg et al., 2015). However, Lusthaus's (2019) research revealed that some small forums are invitation-only communities where a reference from an existing forum member is required for entrance and users in this kind of forums are more likely to commit a crime.

In addition, the technical level of forum users varies greatly. In Holt et al.'s (2012) research, hacker forum members are grouped into three categories according to their technical merit. Most forum members represent learners who are passionate about technology, but lacking the background knowledge needed to successfully conduct a cyber attack. Therefore, they choose the right forum to learn and improve their knowledge of certain technologies. The second category, which has significantly fewer members than the first group, represents individuals who have sufficient skills to understand and use the information shared on these platforms. The most important group brings together the highly skilled hackers but their percentage is usually limited to single digits. They can not only understand and use existing tools and technologies, but they can also create new methods that can be exchanged with others.

One important indicator to measure technical skills of hackers is reputation score, which is measured by user's engagement and other users' feedback. However, reputation measures more than just skills but also user's trustworthiness rate (Allodi et al., 2015). Higher reputation brings more advantages in underground trading market for hackers because they seems to be more credible than others. This economic benefit motivates hackers to reply with more detailed posts and thereby providing more practical resources for newbies and reducing the technical requirement of conducting an attack (Zhang and Li, 2013). This kind of exchange may be one of the possible reasons for the great success of hacker forums. Allodi et al. (2015) has proved that the lack of proper reputation score calculation methods and enforcement of rules were main contributors to the fail of some stolen cards marketplace. They also mentioned that one of the reasons why hackers moved from chat based towards forums is to gain reputation.

2.2 Hacker Identification

Computational approaches such as social network analysis (Lu et al., 2010; Holt et al., 2012; Pastrana et al., 2018) and clustering analysis (Abbasi et al., 2014; Pastrana et al., 2018; Kigerl, 2020) were commonly used to identify hackers in previous research. Some typical literature was reviewed to identify research gaps and guide research design for this essay.

Social-network based methods need to construct a friendship network with a node representing a hacker and weight of the edge between two nodes representing the strength of their relationship. Lu et al. (2010) collected 115 text-based cybersecurity reports such as USA Today and The New York Times from LexisNexis Academia and Google. Then named-entity recognition was applied to retrieve hackers and their relationships to build a hacker network. They simply calculated four measures of centrality of this network for each hacker to rank them and then get four ordered hacker lists in which top hackers are identified as key hackers. Lu et al.'s (2010) paper may be relatively initial research to introduce social network analysis for identifying key hackers. Holt et al. (2012) made some improvement on this basis. They utilized a sample of 336 individuals from a Russian social networking site. They not only constructed a network of individuals but also a network of groups. Furthermore, they introduced a risk index for each hacker based on their creation, distribution and use of malicious software. In the end, not only did they identify key hackers and key hacker groups, they also marked them with risk levels. However, there is still no idea about hackers' interests. Pastrana et al. (2018) built their social network with more metrics to measure hacker's importance such as total number of replies and they also developed tools a to analyse the interests of forum members.

Although social network analysis helps to identify key hackers, it can not provide an overview of all hackers that includes their activities and roles. Other studies have tried to apply statistical clustering techniques to data obtained from hacker forums. Clustering methods aim to take the attributes related to a subject as inputs and group all subjects into multiple different categories based on the shared similarity between the input attributes. Abbasi et al. (2014) identified three categories of hacker features from hackers' posts: the appearance of cybercriminal assets, the usage of hacker specific terminology and forum involvement. Then they implemented the X-means method to cluster all hackers into four categories: black market activist, founding members, technical enthusiast and average users. The result showed that about 86% users were average users with low involvement and only 12% of them were labeled technical enthusiast who often embed code and use hacker terminologies in their posts. Pastrana et al. (2018) introduced two more categories of features into their clustering model: social network features and reputation measures. They group all users into five categories and calculated the centrality for each category but they did not explain hacker's specialties and focuses in these categories. In order to provide a more detailed examination on the final categories, Kigerl (2020) utilized two phases of clustering methods to cluster users into several categories and conducted qualitative analysis on each category to further shed insight into the nature of the forums. In the first phase, Kigerl (2020) used topic model and clustering to extract seven topic features. Together with 13 observed features which were borrowed from previous research, Hierarchical agglomerative clustering was implemented to cluster users on the 20 features in the second phase. Finally, hackers were clustered into 16 categories and these categories were further grouped into four domains: general consumers, location-based consumers, producers, and other users by qualitative analysis. Kigerl's (2020) research introduced more features to cluster hackers, which provided more angles to assess hackers' interests and specialties and made their conclusion more reliable.

2.3 Temporal Analysis on Hacker Community

Understanding how hackers change their interests and roles over time is also important for cybersecurity which allows society to consider ways to prevent potential crime. Past researchers has begun to use temporal analysis to explore hackers' evolution pattern.

Benjamin and Chen (2014) conducted survival analysis to measure how hackers' participation behaviors affect their survival time. Their research provided an method which can be used to predict the trajectory of users in the hacker community, and can help predict which users will participate on a long-term basis. From another point of view, Fang et al. (2016) built a research framework to investigate topics evolution in Chinese hacker communities. They utilized topic model to extracted five topics: Trading, Fraud prevention, Contact for cooperation, Casual chat and Intercepting and monetizing. For each topic, they examined the change of the top 10 key words and manually summarize the trend of change. Fang et al. (2016) explored the changing trends within each topic while Pastrana et al. (2018) investigated the evolution between topics. They first extracted nine interests topics such as market and hacking, etc. and then measured hackers' interests at the beginning, middle and the end of the period each actor has been active. Finally, a visualization was provided to show the transitions of hackers' interests over the three points of time.

2.4 Chapter Conclusion

Throughout this chapter there has been an introduction to the operation of hacker forums and the methods used in past studies for both hacker identification and temporal analysis. Kigerl's (2020) method for identifying hackers' roles was borrowed and improved in this thesis. In order to examine the evolution pattern of hackers' roles, Pastrana et al.'s (2018) idea was implemented and tested on the result of hacker identification.

3 Theoretical Background

This chapter provides an overview of technical background required for readers to better understand this thesis. It begins with introducing the main concepts in the field of machine learning, focusing on the algorithms used in this article. In addition, an overview of the methods used to represent text as feature vectors is provided, including distributed representations and distributed representations.

3.1 Machine Learning

Machine learning is a sub-field of artificial intelligence (AI) (Kononenko and Kukar, 2007), and these two words are often used synonymously. Even so, although the goal of all areas of AI is to introduce intelligence into computer programs, the way in which intelligence is realized in ML is different from other sub-fields of AI. More specifically, the difference lies in the learning process. Machine learning is a data-driven approach that aims to solve problems by using experience from past examples (Mitchell, 1997). In other words, these algorithms are not designed to solve a specific problem, such as algorithms for playing chess or solving Sudoku. A typical example is e-commerce websites, which use machine learning to classify users and items, and use user historical shopping information to recommend products that may be of interest to users to increase sales. Another easy-to-understand example is to use the shape s of the watermelon, the stripe shape ss, the color c, and the sound of percussion p to determine whether a watermelon is good or bad. If there are 10,000 watermelons, among which watermelons showing a certain combination of characteristics $F(s_1, ss_1, c_1, p_1)$ are all good melons. Then if we get a new watermelon which also shows $F(s_1, ss_1, c_1, p_1)$, we have reason to believe that this watermelon is also a good melon. This is the basic idea of machine learning.

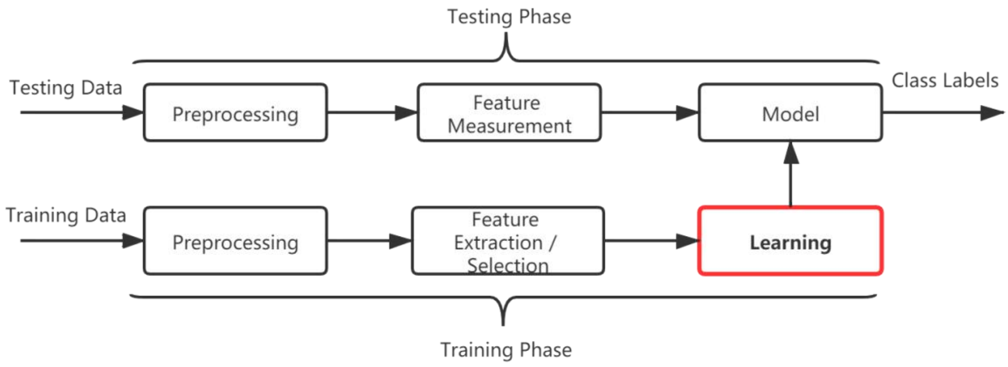


Figure 1. Machine Learning Process (Jain et al., 2000)

Figure 1 shows the main building steps of the machine learning process consisting of two phases: training and testing. In order to make this process better understand, we use the above watermelon example to explain. For example, there are 10,000 watermelons with their information as the raw data and each watermelon is a data sample. In the beginning, they are divided into two groups: training data and testing data for the two phases respectively. The purpose of the training phase is to build a model by the training data which is able to determine whether a watermelon is good or bad. It begins with pre-processing, cleaning and representing the raw data as vectors, where each of the values of the vectors represents some salient feature of the raw data. These features may include the shape, size, color, stripes, weight, etc. of the watermelon, but not all features are useful to judge a watermelon. Therefore, the quality of these features is enhanced by selecting a subset of the features to reduce the dimensionality of the vectors (Feature Selection) or creating new features from the existing ones (Feature Extraction). The first phase is concluded by training the model by a learning algorithm. The second phase is to test the performance of the model built in the first phase on the testing data. Prior to that, the testing data is pre-processed and represented with the same features as during the training. There are currently many types of learning methods, each of which contains many different algorithms. The following paragraphs will briefly introduce three of these methods: supervised learning, unsupervised learning and semi-supervised learning.

3.1.1 Supervised Learning

The significant difference between supervised learning and unsupervised learning is that the data samples for supervised learning must contain output labels, which serve as the "teachers" for the model. These data samples are used to find a relation between features and labels and therefore build a generalized model which is able to label new data samples without output labels. Let each sample X_i be represented by a feature vector $\{x_1,x_2,...,x_n\}$ and $Y_i = \{y_1,y_2,...,y_m\}$ be the set of all m-possible outputs. The objective for supervised learning is to find a function F such that $Y_i = F(X_i)$. Regarding to the type of the output class, two tasks of supervised learning exists: classification and regression. For regression, the output label is continuous, such as predicting house prices based on the characteristics of the house While for classification problems, the output label is discrete, such as judging whether a watermelon is good or bad, judging whether the sentiment of a paragraph is positive or negative, etc.

3.1.2 Unsupervised Learning

Unsupervised learning is the process of inferring the properties of the data without "teachers" as in supervised learning due to the unavailable output labels. The most popular form of unsupervised learning is clustering, which is to group data into different categories so that data within the group are more similar to each other than to data belonging to other groups (Jain et al., 1999). The similarity between data samples can be measured by many metrics such as Euclidean, Manhattan Distance or Cosine dissimilarity, etc. Kmeans clustering (Macqueen, 1967) and hierarchical agglomerative clustering (Calinski, 1974) were used in this thesis and a brief introduction on their process follows.

Kmeans contains three main steps:

1. Randomly initialize K data samples as the K centroids of K clusters.
2. Assign each data sample to the nearest centroid.
3. Recompute each centroid as the mean of data samples assigned to it.

After step 2, all data samples are grouped into K clusters with a centroid for each of them. Then the final result are returned by repeating step 2 and 3 until there is no change in centroids.

Kmeans needs to manually choose the size of K at the beginning, but HAC (hierarchical agglomerative clustering) does not need. At the beginning, it treats each data sample as a cluster, and then combines the two most similar data sample to form a new cluster, and so on until all the data samples are grouped into one cluster. HAC aims to build a cluster dendrogram, a example of which is shown in Figure 2. It is easy to get result for different number of K by dividing this dendrogram.

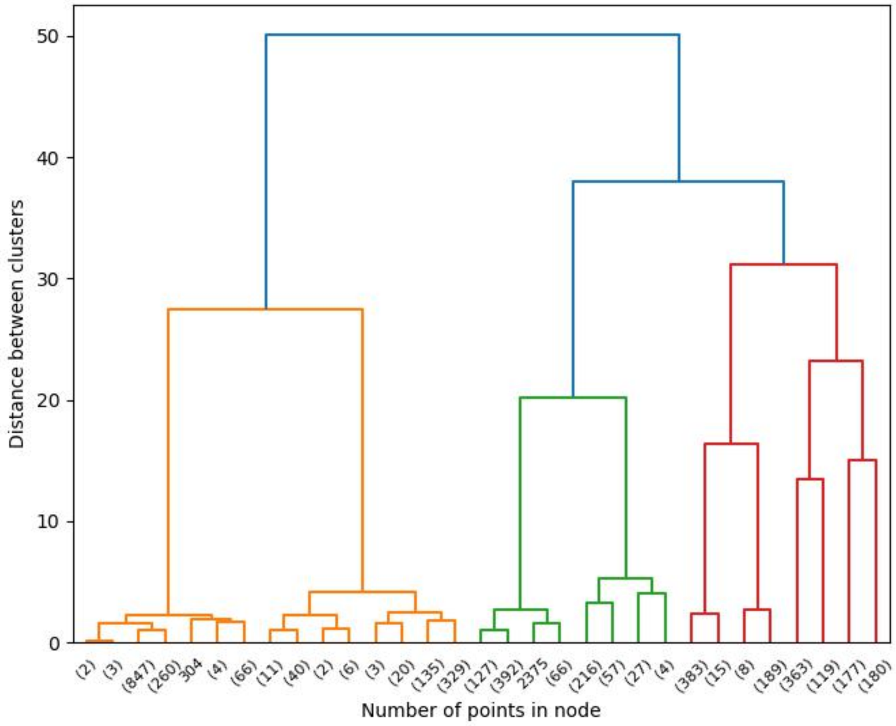


Figure 2. An example of the result of HAC

3.1.3 Semi-supervised Learning

In many practical problems, the cost of labeling data is sometimes high, but a large amount of unlabeled data is easily available. Therefore, semi-supervised learning methods were created combining supervised learning and unsupervised learning, which uses a small amount of labeled data samples to guide and predict the labels of unlabeled data samples. In this thesis, two semi-supervised learning methods were used: LPA (label propagation algorithm) (Zhu and Ghahramani, 2002) and LSA (label spreading algorithm) (Zhou et al., 2004).

LPA begins with constructing a fully connected graph with each data sample as a node. The edges are weighted to define the similarity between nodes, with higher weights illustrating higher levels of similarity. Then, the labeled nodes are used to propagate information to all of the other nodes where larger weights between nodes enables this propagation to occur more readily. Finally, a label probability distribution is calculated for each unlabeled node and the label for each of them is chosen to be the one with highest confidence. LSA is quite similar to LPA, but instead of using graph Laplacian for propagation, it uses normalized graph Laplacian.

3.2 Text Representation for Machine Learning

The data used for machine learning algorithms should be processed and represented in the form of feature vectors. However, the data used in this thesis is mainly in textual format, which should be converted into features vectors. Text representation serves as the role of extracting features from textual documents. In this section, the main principles of the two methods used in this thesis are summarized: TF-IDF (term frequency and inverse document frequency) (Salton and Buckley, 1988) and the mean of Word2vec (Mikolov et al., 2013).

3.2.1 TF-IDF

TF-IDF is a statistical method used to evaluate the importance of a word to one of the documents in a corpus. The main idea of TF-IDF is if a word appears in an document with a high frequency of TF and rarely appears in other documents of the corpus, it is considered that the word or phrase has good classification ability and is suitable for classification.

TF-IDF is actually TF * IDF. A vocabulary first records all words that have appeared in the corpus. For each word in the vocabulary, if $n_{i,j}$ represents the number of occurrences of word i in document j, the $TF_{i,j}$ of word i for document j can be calculated as:

$$\mathrm{TF}_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

If |D| represents the total number of documents in the corpus, d_j for document j and w_i for word i, the IDF_i of word i can be calculated as:

$$\mathrm{IDF_i} = \log \frac{|D|}{|\{j : w_i \in d_j\}|}$$

Then, the $TF\text{-}IDF_{i,j}$ of word i for document j is calculated as $TF_{i,j}$ * IDF_i. After this, a vector v_j of size V (vocabulary size) is created to represent document j in the corpus, the entries of which are set to the TF-IDFs of all the words in the vocabulary for the document j in a fixed order.

3.2.2 The Mean of Word2vec

Word2vec is one methods of the distributed representations (also known as word embedding) which can map a word onto a real-valued vector, whose dimensionality is low compared to TF-IDF, and the columns of which contain the semantic meaning of the word (Turian et al., 2010). Then the mean of all the word vectors in a document is calculated as the representation of the document. Contrary to TF-IDF, Word2vec model is built through an artificial neural network. It has two training models: Skip-gram and CBOW. The CBOW model uses n words before and after the word w to predict the current word w; while the Skip-gram model is the opposite, which uses the word w to predict n words before and after it. From a large number of experimental results, the training speed of the CBOW model is better than the Skip-gram model, but the performance of the Skip-gram model is better.

4 Methodology

This chapter will give the details of the research design from data preparation, model selection to final analysis. Figure 3 shows an overview of the research design, which consists of s series of steps involving automated or semi-automated data processing methods and analysis.

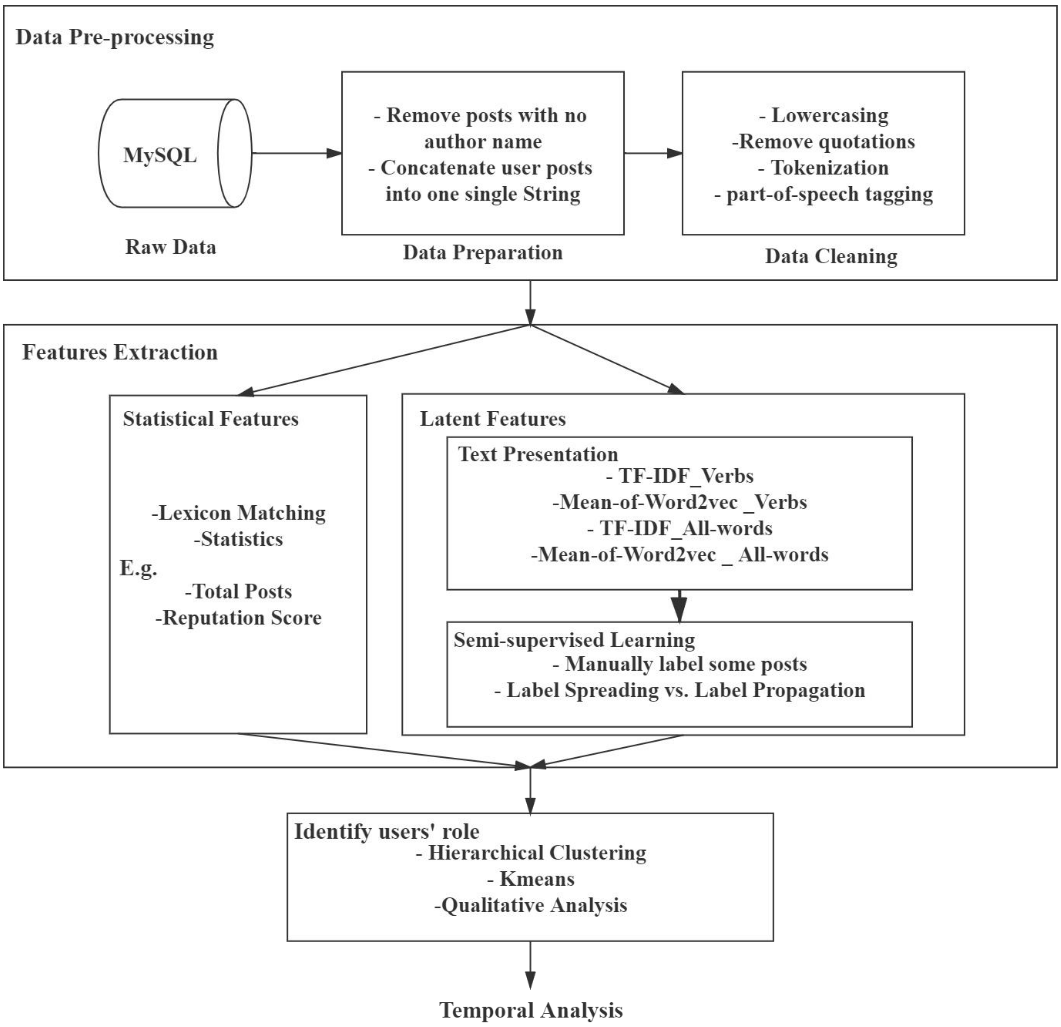


Figure 3. An Overview of Research Design

In the beginning, a hacker forum dataset containing user posts and profile data was acquired for this research. The data was then cleaned and processed for feature extraction.

Next, two types of features were extracted from the processed data: statistical features and latent features. Statistical features reflect the user's participation, technical level and reputation in the forum, which are easy to get by lexicon matching and statistics. E.g. "total posts" and "total hacking terms used". Latent features are actually initial role tags of the users, reflecting their specific behavior in the forum such as "advice giver" and "information sharer", which were extracted by semi-supervised learning.

In addition, two clustering technology (Hierarchical clustering and Kmeans) were utilized to divide users into different role categories based on the combined information of statistical features and latent features. Following categorization, qualitative analysis was conducted to further shed insight into the nature of the user base in this forum.

Finally, a temporal analysis was designed to examine the transitions or evolution patterns of roles of users on this forum.

4.1 Dataset

4.1.1 Data Source

The data were sourced from a hacker forum database called *Exetools* (Schweitzer, 2018), one of the oldest forums that has been active in exchanging hacker assets since 2002. This forum has 24663 posts dating from 2002-1-16 to 2018-3-14, and the ratio of the number of posts to the number of users in this forum is very high compared to the other forums, which makes it suitable for studying longitudinal threat landscape. Apart from posts, users' labels such as membership, join date and reputation, etc were also collected in this dataset, which can be helpful to identify a user. On the other hand, this forum runs like a technical community, in which users are very keen to discuss technical issues and spread hacker assets, thereby providing new comers a platform to learn hacking knowledge. This character makes it suitable to do temporal analysis by detecting evolutionary patterns of hackers.

4.1.2 Data Pre-processing

The acquired file from the online dataset is MySQL database dumps, which was imported into MySQL database afterwards for further cleaning and analysis. After eliminating posts with no author name and removing duplicates, the table retains 771 registered users with 16359 posts dating from 2003-1-1 to 2017-12-31.

In order to facilitate temporal analysis, all the posts were sorted into time intervals with quarters as the unit. In each interval, the textual content of posts of each user were concatenated into one single string for features extraction. This means in every quarter from the beginning date of the dataset, one user can only have no more than 1 concatenated posts which corresponding to a role. After this step, the table retains 771 users with 4053 posts dating from the first quarter of 2003 to the fourth quarter of 2017.

In addition, the letters of all posts were lowercased for further processing such as lexicon matching. Word tokenization was also done in this step for building text representation and part of speech is also labeled for every words to facilitate the word selection for text representation. Furthermore, Some users quoted others' answers when posting, and such quotes were also recorded in the user's post together with the user's own words. In order to better distinguish users, such quotes were removed from the posts. One example of such posts can been below:

Quote: Originally Posted by professor.frink this feature sometimes crashes DIE due to unhandled exception*....Can you provide a file which causes this crash and tell about OS so I could give it to the author to fix it?.*

4.2 Features Extraction

Features helps to describe users in multiple dimensions. In this research, 22 features were defined based on either manual observation of the data or the suggestions in past studies, which consists of 11 statistical features and 11latent features. All these features were extracted using automated or semi-automated scripts that sifted through the dataset and were used to be the dimensions on which to cluster users into different categories.

4.2.1 Statistical Features

The definition of the 11 statistical features can be seen below:

① *Total Posts.* The number of forum posts was counted for each user in each quarter. Forums posts consists of newly created posts that begin a separate thread that users may participate in and responses or replies to such original posts. This feature was used to examine user engagement, which means users who post more posts can be seen as more active users.

② *Total Replies.* The number of replies was also counted alongside total posts. In hacker forums, some users tend to begin new threads to advertise new tools or seek suggestions of a specific topic while others are just keen to participate in threads made by others and give advice. Therefore, this feature can help to distinguish this two kinds of users. There is a “post sequence” number for each posts in a thread. Number 1 means the post is the first post of the thread, otherwise is replies to the original post.

③ *Active Days.* Each forum post has a post date associated with it. Users were labeled active when they posted posts at that day. The number of days user are active were counted as active days, which was used to identify long-term active users and short-term active users.

④ *Total Threads.* The number of total threads user were involved in was counted for each user in each quarter. This feature aims to see if a user only focused on one topic or had a wide range of interests.

⑤ *Average Message Length.* This feature also aims to measure user’s participation. Users who send more words in one post on average may tend to have higher engagement.

⑥ *Total Quotations.* The number of quotations in posts was counted for each user in each quarter. Users who makes a post with a quotation means this post is a response to a specific user. This feature aims to track such interactions.

⑦ *Reputation.* The hacker website calculates a reputation score for each user based on the user's contribution to the forum and participation, which can be used to measure the user's loyalty to the forum.

⑧ *Vouched Status.* In this forum, each user has a designate title attached to their profile which was recorded in the “Membership” field in the database table. Users with titles such as “VIP”,”Founder”,”Developer”,”Cracker” were considered as top ones in the hacker forum pyramid compared to others. A binary variable was used to indicate this feature: 1 represents the presence of any of these titles, 0 represents the opposite. Table 1 Shows the map from titles to vouched status and the count of users and posts for both status 1 and status 0. It is noticeable that 15% vouched users created 31% posts, which turns out that vouched users are more active.

Table 1. Vouched Status Labeling Result

Titles	Vouched Status	Count & % of Posts	Count and % of Users
VIP, Leader, Game Tech, Drunken Squirrel, Founder, Senile Member, Musician Member, Super Moderator, Developer, Retired, Exetools Team Manager, Administrator, Co-Administrator, Moderator	1	1263/31%	118/15%
Friend, Family	0	2790/69%	653/85%

⑨ *Attachment.* Users can attach attachments when posting, which are very likely to be widely spread as hacker assets. The number of attachments associated with posts was counted to track this.

⑩ *Total Coding Terms Used*. Users who use more coding terms in their posts may have a better coding knowledge. Therefore, the number of coding terms used in posts was counted to distinguish technical and non-technical users. This feature was extracted by lexicon matching. A valid coding term list (Hackterms, 2018) was acquired from https://www.hackterms.com/about/all, which contains 13016 coding terms such as “git”, “java”, “cmd”, etc.

⑪ *Total Hacking Terms Used*. Just like coding terms, users who use more Hacking terms in their posts may have a better hacking knowledge. This feature is extracted to distinguish hackers of different levels. A valid hacking term list (Techopedia, 2020) was acquired from https://www.techopedia.com/dictionary/tags/hacking, which contains 2878 hacking terms such as “cracker”, “proxy hacking”, “sql injection”, etc.

In conclusion, Among the 11 features, the first 6 tend to reflect the user’s participation and activity, the next two correspond to the user’s membership, and the last three reflect the user’s technical level.

4.2.2 Latent Features

Latent features are initial role tags of users reflecting their activities in the forum. In this part, machine learning techniques were introduced to extract latent features from users textual posts. While supervised learning requires huge amounts of labeled data and it is not easy to define the final result of unsupervised learning, semi-supervised learning was used in this section, which only needs a small portion of labeled data. After semi-supervised learning, each user was labeled an initial role as the latent feature. Before semi-supervised learning, text representation was required to transfer texts to vectors to feed the machine learning model.

4.2.2.1 Text Representation

One distributional model and one distributed model were utilized in this part: TF-IDF (term frequency and inverse document frequency) and mean-of-Word2vec (mean of Word2vec). Due to the fact that this research focuses on user’s roles and verbs in posts may have a better reflection on this, the above two models were also calculated by only verbs. Finally, four methods of text representation were calculated in total:

(1) TF-IDF_All-words (term frequency and inverse document frequency by all words)

(2) TF-IDF_Verbs (term frequency and inverse document frequency by only verbs)

(3) Mean-of-Word2vec_All-words (mean of Word2vec by all words)

(4) Mean-of-Word2vec_Verbs (mean of Word2vec by only verbs).

4.2.2.2 Semi-supervised Learning

In order to conduct semi-supervised learning, 583 out of 4053 posts were manually labeled, in which 183 posts were used as test data and 400 posts as training data. 11 role labels were defined in the process of manually labeling, which are shown below with descriptions for each label and Table 2 gives the number and proportion of each label with an example following.

① *Advice Giver.* Advice givers tend to answer other users’ questions and give specific suggestions.

② *Resource Giver*. Resource givers tend to give online resources when answering questions such as: e-books, hyper links, online datasets and even source code etc.

③ *Information Sharer*. Information sharers are keen to share the latest hacker information, such as certain software updates and solutions to certain problems, etc.

④ *Resource Sharer*. Resource sharers are very similar to resource giver, but they are more like advertisers to deliver online resources to all users.

⑤ *Advice Seeker*. Advice seekers are peoples who often ask for others' help to give a solution to a problem.

⑥ *Information Seeker*. Information seekers are slightly different from advice seekers. They tend to focus on non-technical issues while advice seekers are more likely to post questions related to programming.

⑦ *Resource Seeker*. Resource seekers tend to seek online resources.

⑧ *Developer*. Developers demonstrate a higher level of technology than other users. They often release updated details of the products they participated in.

⑨ *Reviewer*. Reviewers often evaluate other people's work or products, and their posts often contain emotional expressions such as criticism or praise.

⑩ *Thanks Giver*. Thanks givers correspond to users who often post thanks or express blessings.

⑪ *Trading Participant*. Trading participants are people who share information about buying or selling hacker assets. This kind of role is very rare in the labeled data, only 2 out of 583.

In the labeled data, the developers and advice givers show a higher level of hacking skills and they will explain the problem in more detail, while Trading Participants, information sharers, resource sharers and resource givers seem to have more hacking assets and they are more likely to participate in the management of the hacker community.

Advice givers and advice seekers occupy a large portion of the labeled data, ranking first (38.42%) and second (15.95%) respectively, and only 0.34% users in the labeled data were labeled trading participant. This may indicates that this forum is more like a technical community rather than a trading forum and most users in this forum were communicating technical issues and learning hacking skills.

Table 2. Examples for Each Role Label.

Role Label	N/%	Example
Advice Giver	*224/38.42*	Your best bet, in my opinion, would be emitting different recognizeable byte sequences using...
Resource Giver	*9/1.54*	try this Code: http://www.rifup.tk/guestbook Best Regards.i have same problem for ESET !.BR.with best wishes for all in new year merry Xmas BR.
Information Sharer	*6410.98*	New version out!.ProtectionId v 6.85 (December 2016) ...
Resource Sharer	*40/6.86*	This is the original layout pdf-version of the book, not a e-pub converted one..http://minfil.org/j2hcedb1bf/B_4888-13.pdf.
Advice Seeker	*93/15.95*	Hi everybody, I recently came across a program packed with "ASProtect 1.23 RC4 - 1.3.08.24 - Alexey Solodovnikov" and I tried to unpack it manually.
Information Seeker	*14/2.40*	Is this going to remain private or can you see it going open source in the future?
Resource Seeker	*37/6.35*	For some reason i cannot download the attachment..maybe some admin could help ?
Developer	*111.89*	Bugs fixed and output improved ....https://s31.postimg.org/9t4ixicy3/2016_07_01_141602.jpg.Send me a message in private to discuss this problem..
Reviewer	*406.86*	It's a great poject..It's always good to see someone working actively on an x64 debugger who also responses/includes community feedback..
Thanks Giver	*49/8.40*	Hi, thanks for letting me into this great forum..Special thanks to besoeso..
Trading Participant	*2/0.34*	Yes, many vendors along with the owners of decryptum.com are selling various password cracking software but all those are just try&error type..

Two commonly used semi-supervised learning algorithms called Label Spreading Algorithm (LSA) (Zhou et al., 2004) and Label Propagation Algorithm (LPA) (Zhu and Ghahramani, 2002) were utilized in this part. Together with the four text representation methods, eight combinations were produced. In order to filter out the best combination, accuracy score was introduced to evaluate the performance of these models. All the eight combinations were trained and tested with training data and test data respectively, and then accuracy score was calculated on the test data for each model. The result of model who performed better were used for further clustering.

4.3 Clustering

Hierarchical agglomerative clustering (HAC) (Calinski, 1974) and Kmeans method were borrowed from past research (Kigerl, 2020) to conduct clustering on the 22 features of 4053 concatenated posts. Both methods would not select the best K size on their own. Therefore, fitting test must be conducted and compared for each method to choose the best K value from a successive K value candidate list. Hierarchical clustering can obtain all results of different Ks at one time by building a cluster dendrogram. It is easy to get result for different Ks by dividing this dendrogram. Changing the number of clusters does not need to calculate the attribution of the data points again. While Kmeans needs to recalculate the attribution of all points for different k sizes. Two through 35 K sizes were attempted and two measures of fit were calculated to evaluate the performance for different clustering methods with different Ks. Each method is an internal clustering model fitting metric used to evaluate the similarity of documents assigned to the same topic and the degree of separation between each topic and all alternative topics. The first one is called DBI (Davies-Bouldin Index) (Davis and Bouldin, 1979), which is a minimization fit metric where lower score indicates a better performance. While the other one is a maximization metric called MSC (Mean Silhouette Coefficient for all test data) (Rousseeuw, 1987) where higher score represents a better fit. The appropriate k size was determined for each clustering method by the visualization of each metric and the result of the better one was delivered to qualitative analysis. After all the posts were grouped into k categories, qualitative analysis was conducted to examine the characteristic and define a role name for each category according to the centroid of features.

4.4 Temporal Analysis

Temporal analysis was conducted to examine the evolution pattern of hackers on the result of 4.3. In this step, the entire duration of the posts from the first quarter of 2003 to the fourth quarter of 2017 was split into several time periods, The change or evolution of hackers' role across these time periods was tracked for each user and all changes were aggregated to see the overall evolution pattern. More details of the design of temporal analysis and results were shown in 5.4.

5 Results

This chapter provides details of the experiments performed according to the research design and delivers details of the result and analysis. Most of the processes was implemented by Python scripts, and all the tools and software used in this thesis are open source and freely available on the internet. Table 3 gives a list of requirements used in the experiments.

Table 3. Requirement for Experiment

Tool	Purpose
Python 3.7.4	Run all the scripts
Sklearn 0.21.3	Text Representation (TF-IDF), Semi-supervised learning and Clustering
Gensim 3.8.1	Text Representation (Word2vec)
Numpy 1.16.5	Mathematical calculations
Pandas 0.25.1	Data Pre-processing and Computing
Matplotlib 3.1.1	Visualization
Plotly 4.10.0	Visualization
MySQL Community Server	Data storage

5.1 Text Representation and Semi-supervised Learning

Two semi-supervised learning algorithm were conducted to extract latent features from the textual content of user’s posts in this research: LSA (Label Spreading Algorithm) and LPA (Label Propagation Algorithm). In order to feed this two model, all the posts were converted into vectors by four text representation methods: (1) TF-IDF_All-words, (2) Mean-of-Word2vec_All-words, (3) TF-IDF_Verbs and (4) Mean-of-Word2vec_Verbs. The four text representation models were all tested by the two semi-supervised leaning algorithm respectively on the test data. Table 4 records the accuracy score for each combination. It is clear to see that LSA with Mean-of-Word2vec_All-words yields better performance.

Table 4 Result of the 8 combinations

Accuracy	TF-IDF		Mean-of-Word2vec	
	All words	Only verbs	All words	Only verbs
LSA	0.4098	0.3934	**0.6885**	0.6230
LPA	0.3934	0.4262	0.3661	0.4426

5.2 Features Exploration

There are 11 statistical features and 11 latent features extracted in the feature extraction part of the methodology. This part delivers an overview of the extracted features.

5.2.1 Statistical Features

Statistical features are related to users' engagement, reputation and technical merit. Table 5 shows the descriptive statistics of these features, among which the first six features indicate users' participation, the next two are connected with users' status and last three reflect users' technical level. As can be seen in the table, users in *Exetools* participated in 3 threads posting 4 posts per quarter on average. The means of "Average Message Length" and "Reputation" reach 64.13 and 59.53 respectively, but the difference between each observation and the overall mean is large (Std: 112.57 and 142.24). On the other hand, the magnitudes of different features are quite different, some have an average value below 1 (Attachment: 0.2), and some reach 64.13 (Average Message Length). In order to eliminate such gap and make different features have the same importance, all features are scaled to the range of 0 to 1. Concretely, for each feature f of each concatenate post p, the normalized feature value $N_{(f,p)}$ is calculated from the original value $O_{(f,p)}$ and the max (Max_f) and min (Min_f) value of the feature f as:

$$\mathrm{N}_{(f,p)} = (O_{(f,p)} - Min_f)/(Max_f - Min_f) \qquad \text{for each f of each p}$$

Table 6 gives the descriptive statistics of normalized features.

Table 5. Descriptive Statistics for Statistical Features before Normalization

Features		%/Mean (Std)
Engagement	Total Posts	4.04 (7.76)
	Active Days	3.24 (4.81)
	Total Threads	2.72 (4.13)
	Total Replies	3.61 (7.23)
	Total Quotations	0.80 (2.36)
	Average Message Length	64.13 (112.57)
Status	Vouched	31.16
	Reputation	59.53 (142.24)
Technical Merit	Attachment	0.20 (0.83)
	Total Coding Terms Used	14.64 (40.86)
	Total Hacking Terms Used	0.36 (1.56)

Table 6. Descriptive Statistics for Statistical Features after Normalization

Features		%/Mean (Std)
Engagement	Total Posts	0.0143 (0.0364)
	Active Days	0.0325 (0.0697)
	Total Threads	0.0210 (0.0504)
	Total Replies	0.0179 (0.0360)
	Total Quotations	0.0204 (0.0605)
	Average Message Length	0.0146 (0.0265)
Status	Vouched	31.16
	Reputation	0.0470 (0.1122)
Technical Merit	Attachment	0.0079 (0.0331)
	Total Coding Terms Used	0.0125 (0.0349)
	Total Hacking Terms Used	0.0103 (0.0445)

5.2.2 Latent Features

Each latent feature for each concatenate post is a role label indicating the user's activity and each post only has one label. Table 7 shows the distribution for latent features. It is clear that advice givers and advice seekers account for a large proportion, more than half of the total, which means that most users tend to discuss technical issues on this forum. About a quarter of users are sharers who take advantage of this forum to advertise malware, spread hyperlinks and e-books and share news in hacker community, etc. Trading participants only account for 0.05%, which indicates that *Exetools* appears to be a technical community and there are very few posts talking about software trading. Trading behaviour may occur in private messages instead of public posts which also verifies previous research.

Table 7. Distribution for Latent Features

Features	%
Advice Giver	42.66
Advice Seeker	14.68
Thanks Giver	11.89
Information Seeker	0.30
Information Sharer	21.96
Developer	0.32
Resource Giver	1.46
Resource Seeker	1.28
Resource Sharer	4.37
Reviewer	1.04
Trading Participant	0.05

5.3 Clustering Results

HAC (Hierarchical agglomerative clustering) and Kmeans were applied on the features of 4053 concatenate posts to divide them into different categories. Fitting tests for each of 2 through 35 cluster size selections for both clustering methods were conducted to filter out the optimum k cluster size. Two fit metrics were plotted for each iteration composed of MSC (Mean Silhouette Coefficient) and DBI (Davies-Bouldin Index). MSC is a maximization metric while DBI is a minimization metric. The result of the fitting tests are shown in Figure 4. It is interesting to see from Figure 4 that when k = 19, MSC reaches its maximum value and DBI just at the bottom of the curve, which is true for both HAC and Kmeans. Therefore, a K size of 19 was selected and each concatenate post was thus assigned a category code from the 19 available categories by HAC and Kmeans.

Qualitative analysis was then conducted to label each category according to the performance of the 22 features. The 19 cluster were further qualitatively grouped into five different domains: Producers, Providers, Teachers, Learners and Others. Due to the fact that Kmeans performed better on the two metrics, the result of the Kmeans was thus used in qualitative analysis. The group descriptive statistics of the five domains are shown in Table 8-12, with a qualitative examination following for each table. In order to make it more convenient to observe the difference of features between each category and the overall average level, for each feature f, a normalized mean feature value $N_{(f,c)}$ of each category was calculated from the original mean feature value $M_{(f,c)}$ and the total mean $M_{(f)}$ as:

$$N_{(f,c)} = (M_{(f,c)} - M_{(f)}) / M_{(f)} \qquad \text{for each f of each c}$$

Therefore, in Table 8-12, a $N_{(f,c)}$ greater than 0 means that this feature is above the overall average level.

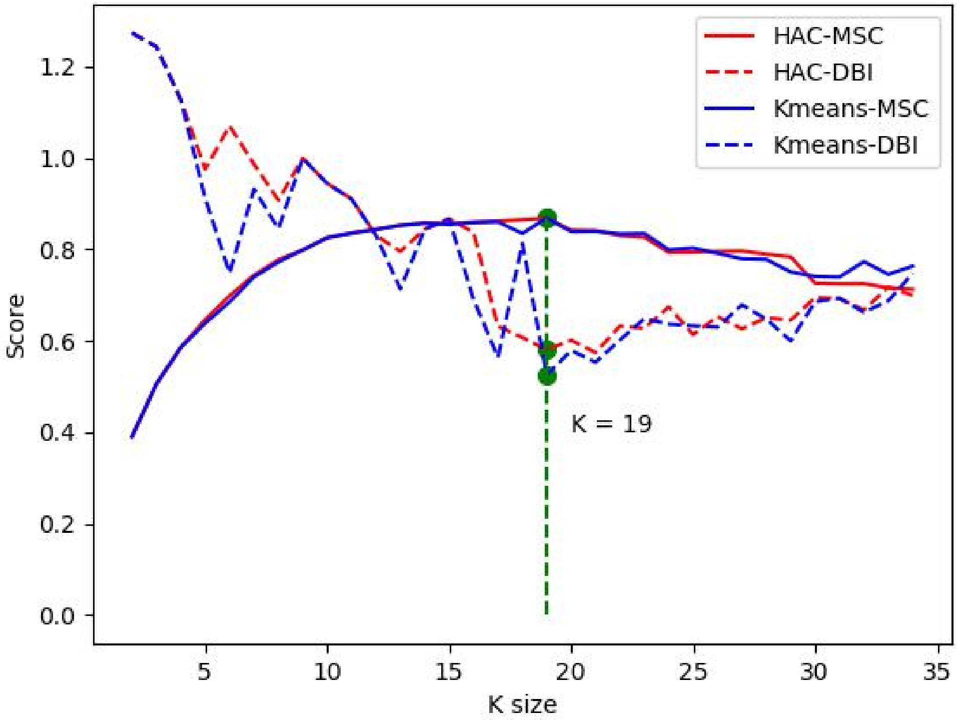


Figure 4. Result of Fitting Tests

Producers

Table 8. Group Descriptive Statistics: Producers (n = 46/1.13%)

Features		%/$N_{(f,c)}$ (STD)	
		Top n = 31	General n = 15
Statistical Features			
Engagement	Total Posts	10.74 (0.22)	-0.14 (0.02)
	Active Days	8.57 (0.29)	0.04 (0.05)
	Total Threads	9.82 (0.26)	-0.34 (0.02)

	Total Replies	8.64 (0.22)	-0.02 (0.02)
	Total Quotations	5.13 (0.13)	-0.16 (0.05)
	Average Message Length	-0.03 (0.01)	1.3 (0.07)
Status	Vouched Status	96.77	40
	Reputation	13.58 (0.29)	-0.25 (0.03)
Technical Merit	Attachment	7.94 (0.18)	-0.66 (0.01)
	Total Coding Terms Used	8.11 (0.17)	1.15 (0.04)
	Total Hacking Terms Used	5.73 (0.12)	3.82 (0.1)
Latent Features			
Role	Advice Giver	0	0
	Advice Seeker	0	0
	Thanks Giver	0	0
	Information Seeker	0	0
	Information Sharer	**100**	0
	Developer	0	**86.67**
	Resource Giver	0	0
	Resource Seeker	0	0
	Resource Sharer	0	0
	Reviewer	0	0
	Trading Participant	0	13.33

Table 8 depicts the two different type of producers: top producers and general producers, who only account for 1.13% in the total dataset.

Top producers are 100% information sharers according to the latent role label they were assigned and almost 100% vouched. The reason this category was grouped into producers is their performance on almost all statistical features is much higher than the average, even more than ten times the average (total posts: 10.74, reputation: 13.58). They have a very high reputation, proficient hacking skills and also actively participate in forum activities. It is no exaggeration to say that top producers are a group of people at the top of the forum’s pyramid. They may be deeply involved in the management and maintenance of the forum, and are more likely to participate in the construction of the underground hacker industry chain.

General producers are 86.67% developers and 40% vouched. Unlike top producers, general producers’ enthusiasm for posting remained at an average level, but they often posted long messages to release their software update information. Interestingly, all developers and trading participants were in the same category, which may reveals that developers are more likely to be involved in malware transactions.

Providers

Table 9. Group Descriptive Statistics: Providers (n = 1036/25.56%)

Features		%/N$_{(f,c)}$ (STD)			
		Senior n = 273	Active n = 586	Inactive n = 47	Temporary n = 130
Statistical Features					
Engagement	Total Posts	1.43 (0.05)	0.31 (0.03)	-0.38 (0.02)	-0.52 (0.02)
	Active Days	1.57 (0.11)	0.28 (0.06)	-0.35 (0.06)	-0.52 (0.04)
	Total Threads	1.48 (0.07)	0.29 (0.04)	-0.35 (0.04)	-0.54 (0.03)
	Total Replies	1.07 (0.04)	0.19 (0.03)	-0.28 (0.02)	-0.41 (0.01)
	Total Quotations	1.59 (0.13)	0.39 (0.06)	-0.55 (0.02)	-0.48 (0.02)
	Average Message Length	0.34 (0.02)	0.44 (0.02)	-0.37 (0.01)	0.01 (0.03)
Status	Vouched Status	100	0	100	0
	Reputation	0.7 (0.09)	-0.53 (0.05)	1.16 (0.11)	-0.63 (0.05)
Technical Merit	Attachment	1.27 (0.05)	0.51 (0.03)	-0.46 (0.02)	0.01 (0.05)
	Total Coding Terms Used	1.29 (0.04)	0.48 (0.03)	-0.48 (0.02)	-0.48 (0.02)
	Total Hacking Terms Used	1.36 (0.07)	0.49 (0.05)	-0.7 (0.01)	0.09 (0.07)
Latent Features					
Initial Role	Advice Giver	0	0	0	0
	Advice Seeker	0	0	0	0
	Thanks Giver	0	0	0	0
	Information Seeker	0	0	0	0
	Information Sharer	**100**	**100**	0	0
	Developer	0	0	0	0
	Resource Giver	0	0	0	0
	Resource Seeker	0	0	0	0
	Resource Sharer	0	0	**100**	**100**
	Reviewer	0	0	0	0
	Trading Participant	0	0	0	0

Table 9 contains the categories of four different types of providers discovered, which make up 25% in the five groups. They are senior providers, active providers, inactive providers and temporary providers who tended to advertise new software, share important community news, spread hacker assets such as source code, e-books, etc. They were labeled as such given 100% of users from this group are either resource sharers or information sharers.

Senior providers are 100% vouched members on *Exetools* who made more posts and were more active than the other three. Active providers are not vouched but their engagement are above the average level, which is the opposite of inactive providers who are 100% vouched but rarely posted. Inactive providers used to be active or even senior providers, but they may have moved to another forum or community, thus reducing the frequency of visits to this forum. Temporary providers were neither vouched members nor active to interact with others. They appeared to create the account just to advise a tool or malware and then became zombie users no longer posting.

Teachers

Table 10. Group Descriptive Statistics: Teachers (n = 1788/44.12%)

Features		%/$N_{(f,c)}$ (STD)			
		Responsible n = 546	Inactive n = 18	Temporary 1 n = 41	Temporary 2 n = 1183
Statistical Features					
Engagement	Total Posts	0.16 (0.03)	-0.34 (0.02)	-0.33 (0.02)	-0.48 (0.02)
	Active Days	0.24 (0.07)	-0.18 (0.05)	-0.23 (0.04)	-0.47 (0.04)
	Total Threads	0.23 (0.05)	-0.26 (0.03)	-0.22 (0.03)	-0.46 (0.03)
	Total Replies	0.15 (0.03)	-0.21 (0.02)	-0.28 (0.01)	-0.37 (0.02)
	Total Quotations	0.16 (0.06)	-0.23 (0.02)	-0.6 (0.01)	-0.34 (0.04)
	Average Message Length	-0.04 (0.02)	0.12 (0.02)	-0.19 (0.01)	0.1 (0.04)
Status	Vouched Status	100	100	0	0
	Reputation	1.37 (0.16)	2.85 (0.25)	-0.71 (0.02)	-0.66 (0.04)
Technical Merit	Attachment	-0.04 (0.03)	2.08 (0.05)	-0.02 (0.03)	-0.45 (0.02)
	Total Coding Terms Used	-0.08 (0.02)	-0.06 (0.03)	-0.32 (0.01)	-0.29 (0.04)
	Total Hacking Terms Used	-0.11 (0.03)	-0.69 (0.01)	-0.59 (0.01)	-0.28 (0.04)
Latent Features					
Initial Role	Advice Giver	**100**	**100**	0	0
	Advice Seeker	0	0	0	0
	Thanks Giver	0	0	0	0
	Information Seeker	0	0	0	0
	Information Sharer	0	0	0	0
	Developer	0	0	0	0
	Resource Giver	0	0	**100**	**100**
	Resource Seeker	0	0	0	0
	Resource Sharer	0	0	0	0
	Reviewer	0	0	0	0
	Trading Participant	0	0	0	0

Teachers accounts for the largest proportion (44.12%) of the five groups. As can be seen in Table 10, four types of teachers were identified based on their latent role:

responsible teachers, inactive teachers and two classes of temporary teachers. Responsible teachers and inactive teachers are 100% advice givers and 100% vouched. The difference in participation distinguishes this two types of people. Responsible teachers were more active in helping questioners so that they seemed to be more “responsible”. Inactive teachers may come from responsible teachers, but for some reason, they gradually reduced the login time. The other two categories were labeled temporary teachers as they both perform badly on statistical features and they both are not vouched. Temporary teachers provided tools as advice which is different from responsible teachers and inactive teachers.

Learners

Table 11. Group Descriptive Statistics: Learners (n = 1077/26.57%)

Features		%/$N_{(f,c)}$ (STD)			
		Outstanding n = 197	General n = 398	Temporary 1 n = 119	Temporary 2 n = 363
Statistical Features					
Engagement	Total Posts	0.97 (0.05)	0.07 (0.03)	-0.94 (0)	-0.95 (0)
	Active Days	0.97 (0.09)	0.03 (0.06)	-0.94 (0.01)	-0.95 (0.01)
	Total Threads	0.89 (0.07)	0.02 (0.04)	-0.93 (0)	-0.95 (0)
	Total Replies	0.73 (0.05)	0.02 (0.03)	-0.69 (0)	-0.7 (0)
	Total Quotations	0.41 (0.06)	-0.06 (0.06)	-0.81 (0.01)	-0.8 (0.01)
	Average Message Length	-0.11 (0.01)	0.02 (0.01)	-0.77 (0.01)	-0.78 (0.01)
Status	Vouched Status	100	0	100	0
	Reputation	1.17 (0.17)	-0.73 (0.02)	0.77 (0.1)	-0.62 (0.05)
Technical Merit	Attachment	0.54 (0.03)	0.01 (0.04)	-0.92 (0.01)	-0.72 (0.01)
	Total Coding Terms Used	0.3 (0.03)	0.14 (0.03)	-0.92 (0.01)	-0.94 (0)
	Total Hacking Terms Used	0.37 (0.04)	0 (0.03)	-0.86 (0.01)	-0.93 (0)
Latent Features					
Initial Role	Advice Giver	0	0	0	0
	Advice Seeker	**100**	**100**	0	0
	Thanks Giver	0	0	**100**	**100**
	Information Seeker	0	0	0	0
	Information Sharer	0	0	0	0
	Developer	0	0	0	0
	Resource Giver	0	0	0	0
	Resource Seeker	0	0	0	0
	Resource Sharer	0	0	0	0
	Reviewer	0	0	0	0
	Trading Participant	0	0	0	0

Advice seekers and thanks givers are grouped into learners, composed of outstanding learners, general learners, and two categories of temporary learners. This group need to be focused because the vast majority of attacks are carried out by participants with a low technical level (Noroozian, 2016).

Outstanding learners are 100% vouched and their reputation is two times of the average. They were active participants for a long time on this forum and used this forum to find solutions. General learners is like new users who are not vouched but seemed to be satisfied with teachers' advice and chose to discuss more questions on this forum. Contrary to the previous two, temporary learners are very similar to temporary providers who appears to be one-time users. They may no longer log in to this forum after solving their problems.

Others

Table 12. Group Descriptive Statistics: Others (n = 106/2.62%)

Features		%/$N_{(f,c)}$ (STD)				
		Reviewer 1 n = 12	Resource Seeker 1 n = 13	Resource Seeker 2 n = 39	Reviewer 2 n = 30	Information Seeker n = 12
Statistical Features						
Engagement	Total Posts	-0.53 (0.01)	-0.52 (0.01)	-0.78 (0.01)	-0.95 (0)	-1 (0)
	Active Days	-0.44 (0.02)	-0.55 (0.01)	-0.78 (0.02)	-0.94 (0.01)	-1 (0)
	Total Threads	-0.37 (0.02)	-0.64 (0.01)	-0.82 (0.01)	-0.9 (0.01)	-1 (0)
	Total Replies	-0.38 (0.01)	-0.4 (0.01)	-0.57 (0.01)	-0.68 (0)	-0.75 (0)
	Total Quotations	-0.06 (0.04)	-0.42 (0.02)	-0.65 (0.02)	-0.87 (0.01)	-0.79 (0.01)
	AML	-0.59 (0)	-0.38 (0.01)	-0.26 (0.02)	-0.7 (0)	-0.5 (0.01)
Status	Vouched Status	100	100	0	0	17
	Reputation	-0.44 (0.03)	-0.21 (0.04)	-0.71 (0.02)	-0.83 (0.01)	-0.44 (0.05)
Technical Merit	Attachment	-0.16 (0.02)	-1 (0)	-0.61 (0.01)	-0.83 (0.01)	-1 (0)
	TCTU	-0.85 (0)	-0.67 (0.01)	-0.57 (0.01)	-0.94 (0)	-0.94 (0)
	THTU	-0.77 (0.01)	-0.79 (0.01)	-0.79 (0.01)	-0.35 (0.01)	-1 (0)
Latent Features						
Initial Role	Advice Giver	0	0	0	0	0
	Advice Seeker	0	0	0	0	0
	Thanks Giver	0	0	0	0	0
	Information Seeker	0	0	0	0	**100**
	Information Sharer	0	0	0	0	0

Developer	0	0	0	0	0
Resource Giver	0	0	0	0	0
Resource Seeker	0	**100**	**100**	0	0
Resource Sharer	0	0	0	0	0
Reviewer	**100**	0	0	**100**	0
Trading Participant	0	0	0	0	0

Five of the 19 categories did not fully belong to the four groups of producers, providers, teachers and learners. The centroids of them were represented in Table 12. They are labeled the same as their initial role: reviewer, resource seeker and information seeker. Users in this domain all had a very low involvement. Some of them used to be activists (100% vouched), but gradually withdrew, while others were just like on-time users.

5.4 Temporal Analysis

Five groups of people were identified in 5.3 and each of them has several sub-categories. This part will conducted temporal analysis to investigate the evolution pattern between five groups and within each groups.

5.4.1 Between Five Groups

To analysis the temporal evolution between groups, users' role was tracked in three time points for each user: start, middle and end. The start was defined as the quarter of their first post, the end was the quarter of their last post and the middle was the period in between. All changes across these three periods were aggregated as the overall evolution.

Figure 5 shows the aggregated evolution over the three periods. At each time point, groups were sorted by the number of users. Overall, most hackers were teachers, providers and learners and the transition seems to be complicated. Over their time in the forum, there was a slight increase of learners and teachers, and a decrease of providers. About 1/3 learners became teachers in the end and a small portion of learners became providers. These two groups of people got more experienced in the forum from the start to the end which may reflect their potential threats.

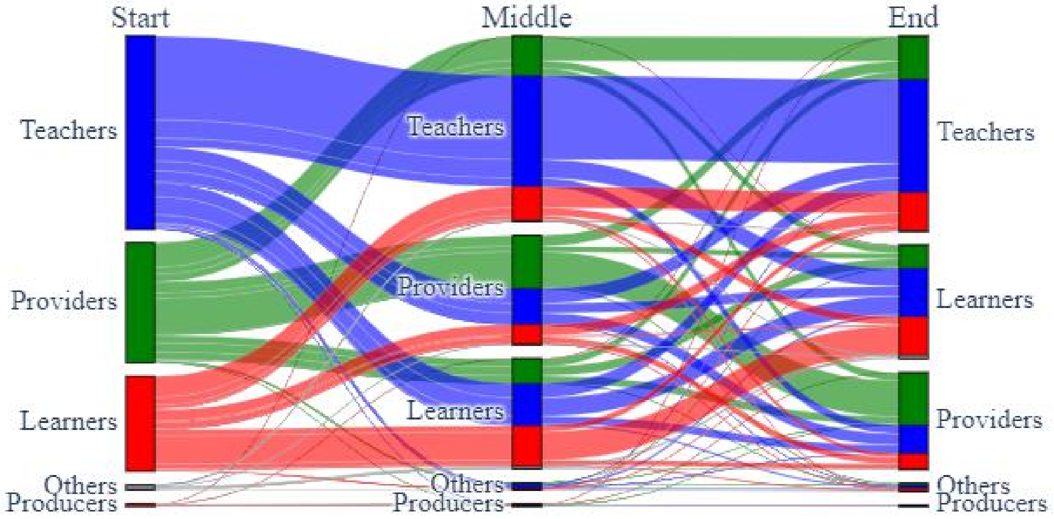
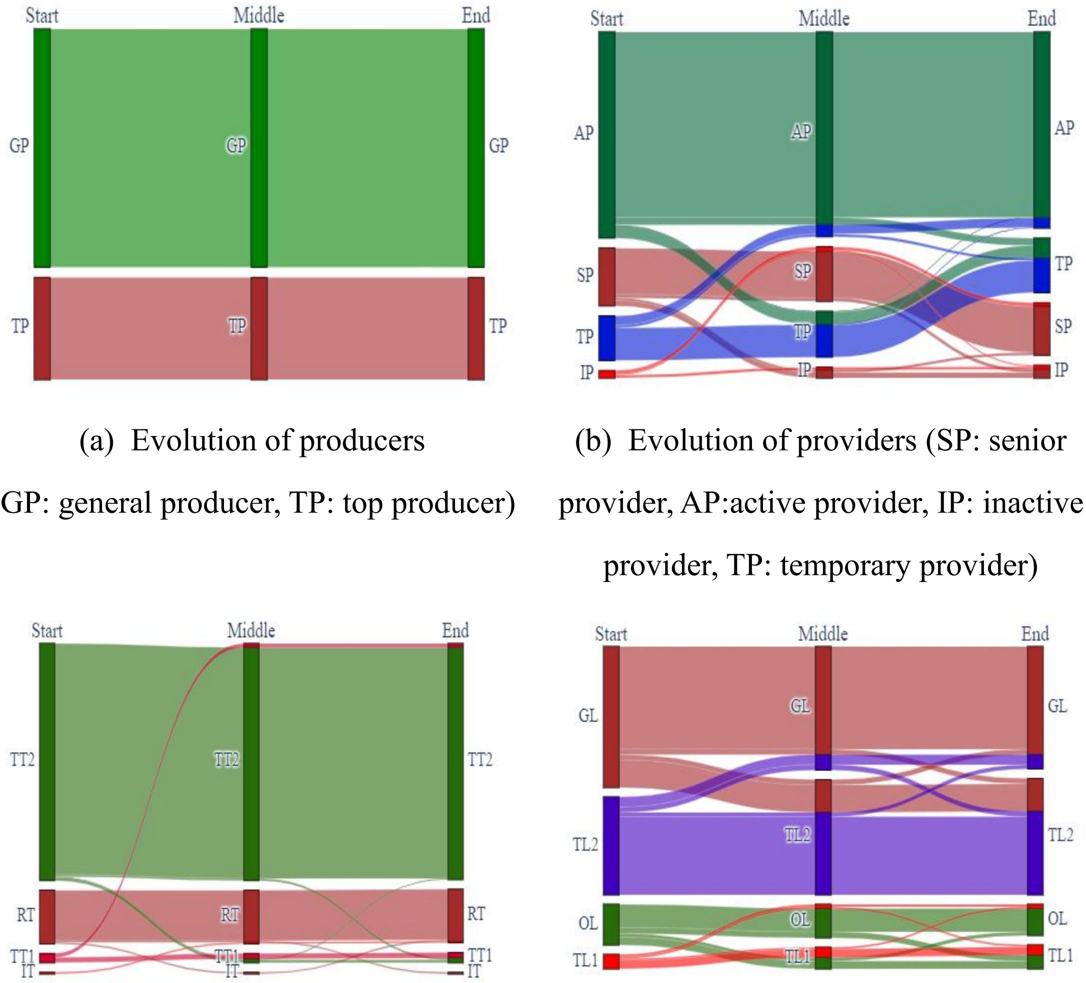


Figure 5. Evolution of roles of hackers from start, middle to end of their activity in Exetools

5.4.2 Within groups

The evolution within each group except others were also conducted using the same method as 5.4.1, which is shown in Figure 6. There seems to be no evolution within producers. Top producers at the start time remained top producers at the end and general producers remained general producers. Meanwhile, there were also very few changes between teachers in each categories. However, interesting transitions were found in providers and learners. Many senior providers in the beginning became inactive providers in the end and some active providers transformed into temporary ones, which may caused by some people moving to another platform. On the other hand, there were a noticeable portion of temporary providers became active in the end. Similar transitions happened within learners where some general learners and outstanding learners tended to become temporary learners and some low level learners became high skilled ones. All this transitions reveals that some users gradually withdrew and their positions were replaced by some new users.



(a) Evolution of producers (GP: general producer, TP: top producer)

(b) Evolution of providers (SP: senior provider, AP:active provider, IP: inactive provider, TP: temporary provider)

(c) Evolution of learners (OL: outstanding learner, GL: general learner,TL: temporary learner)

(d) Evolution of learners (OL: outstanding learner, GL: general learner,TL: temporary learner)

Figure 6. Evolution of roles within groups

6 Discussion and Limitations

6.1 Discussion

This study sought to examine the contents of a hacker forum database containing 771 registered users with 16359 posts dating from 2003-1-1 to 2017-12-31. Two phases of machine learning methods were conducted to group users into meaningful categories. The first phase classified users into 11 latent role topics by semi-supervised learning. The second phase utilized Kmeans clustering to further group users based on the role topics as well as 11 statistical features on user forum activity and status, resulting in 19 categories. Qualitative analysis was done to further group them into five domains: providers, producers, teachers, learners and others. In the end, temporal analysis was designed to track changes of user's role on this forum.

Many of the findings revealed in this research were consistent with prior research on hacker forums. Highly skilled hackers only accounts for a very small part who participate in forum discussion very often and have a high reputation and they may also be involved in the management of the forum. Most users are general users whose technical merit and participation are at average level or even lower than average.

There are also some findings have not been adequately touched in prior research. Some researchers (Sood and Enbody, 2013) divided users into buyers, producers, advertisers, etc. In this study, providers and producers were also identified but there seemed to be no buys on the forum involved in this research. Instead, a new role, learners were identified in this research consisting of four sub-categories. Learners are even more dangerous than teachers as the majority of attacks were committed by low skilled hackers or even newbies(Noroozian et al., 2016). Furthermore, temporal analysis tracked the potential offenders who were low level learners at the beginning and became high skilled ones in the end.

Qualitative analysis provided more details of the five domains while temporal analysis revealed the evolution pattern within each domain and between domains. The number of teachers ranked first on this forum followed by providers and learners. About 50% users are temporary or one-time users, whose participation were very low.

Users who ask for a resource or a piece of news tended to finish their interaction after achieving their goal. The temporal analysis also tracked the transitions of users who used to be a activist but became a temporary user in the end.

6.2 Limitations

This research presented a longitudinal study of roles of hackers in an underground forum and has attempted to overcome the difficulties in hacker identification and temporal analysis. However, a number of limitations remain. First, the forum chosen in this research may not be a typical one, thus the findings uncovered may not be applicable to all forums. In the same vein, users can send private messages to another one on the forum, but features involved in them were not included because the data is not available. Finally, the machine learning algorithms utilized in this research are basic ones which can be replaced by more advanced models.

7 Conclusion and Future Work

The ease of access to attack tools and hacking skills make underground forums attractive places for young, non-skilled people to learn about hacking. This research built a semi-automatic model to identify hackers' role and designed a meaningful temporal analysis to examine the evolution pattern of roles of hackers on a hacker forum. Identifying hackers helps to know who these hackers are while analyzing the evolution of hackers especially low-level hackers make it possible to consider prevention measures. A two-stage process was conducted to group users into 19 categories which were further grouped into five domains by qualitative analysis: producers, providers, teachers, learners and others. Within each domain, users were divided according to their participation, membership and technical level. After this, temporal analysis was designed to track users' evolution over three periods of time. Learners who were thought to be at risk of getting involved in crime were identified in temporal analysis. These users were low level hackers in the beginning but grew into skilled ones or teachers in the end.

The research done in this thesis also has many potential extensions. First, more forums can be tested with the method built in this research to investigate more general findings applicable to most hacker forums. Second, more advanced machine learning algorithms and natural language processing methods can be utilized in text representation such as Glove, ELMo and BERT, etc. Finally, more meaningful features can be introduced in hacker identification part such as social network centrality features and measures related to private messages as new features may uncover more interesting findings.

Bibliography

Allodi, L. (2017). Economic Factors of Vulnerability Trade and Exploitation. Acm Sigsac Conference. ACM.

Lusthaus, J. (2019). Beneath the Dark Web: Excavating the Layers of Cybercrime's Underground Economy. *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE.

Zhang, X., & Li, C. (2013). Survival analysis on hacker forums. In *SIGBPS Workshop on Business Processes and Service* (pp. 106-110).

Karami, M., & McCoy, D. (2013). Rent to pwn: Analyzing commodity booter DDoS services. *Usenix login*, *38*(6), 20-23.

Kigerl, A. (2020). Behind the Scenes of the Underworld: Hierarchical Clustering of Two Leaked Carding Forum Databases. *Social Science Computer Review*, 0894439320924735.

Allodi, L., Corradin, M., & Massacci, F. (2015). Then and now: On the maturity of the cybercrime markets the lesson that black-hat marketeers learned. *IEEE Transactions on Emerging Topics in Computing*, *4*(1), 35-46.

Garg, V., Afroz, S., Overdorf, R., & Greenstadt, R. (2015). Computer-supported cooperative crime. In *International Conference on Financial Cryptography and Data Security* (pp. 32-43). Springer, Berlin, Heidelberg.

Holt, T. J., Strumsky, D., Smirnova, O., & Kilger, M. (2012). Examining the social networks of malware writers and hackers. *International Journal of Cyber Criminology*, *6*(1).

Lu, Y., Luo, X., Polgar, M., & Cao, Y. (2010). Social network analysis of a criminal hacker community. *Journal of Computer Information Systems*, *51*(2), 31-41.

Pastrana, S., Hutchings, A., Caines, A., & Buttery, P. (2018). Characterizing eve: Analysing cybercrime actors in a large underground forum. In *International* *symposium on research in attacks, intrusions, and defenses* (pp. 207-227). Springer, Cham.

Abbasi, A., Li, W., Benjamin, V., Hu, S., & Chen, H. (2014). Descriptive analytics: Examining expert hackers in web forums. In *2014 IEEE Joint Intelligence and Security Informatics Conference* (pp. 56-63). IEEE.

Benjamin, V., & Chen, H. (2014). Time-to-event modeling for predicting hacker IRC community participant trajectory. In *2014 IEEE Joint Intelligence and Security Informatics Conference* (pp. 25-32). IEEE.

Fang, Z., Zhao, X., Wei, Q., Chen, G., Zhang, Y., Xing, C., ... & Chen, H. (2016). Exploring key hackers and cybersecurity threats in chinese hacker communities. In *2016 IEEE Conference on Intelligence and Security Informatics (ISI)* (pp. 13-18). IEEE.

Kononenko, I., & Kukar, M. (2007). *Machine learning and data mining*. Horwood Publishing.

Mitchell, T. M. (1997). Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, *45*(37), 870-877.

Jain, A. K., Duin, R. P., & Mao, J. (2000). Statistical Pattern Recognition: A Review, IEEE Transactions on Pattern Analysis and Machine Intelligence. *vol*, *22*.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, *31*(3), 264-323.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (Vol. 1, No. 14, pp. 281-297).

Caliński, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, *3*(1), 1-27.

ZhuΓ́, X., & GhahramaniΓ́н, Z. (2002). Learning from labeled and unlabeled data with label propagation.

Zhou, D., Bousquet, O., Lal, T. N., Weston, J., & Schölkopf, B. (2004). Learning with local and global consistency. In *Advances in neural information processing systems* (pp. 321-328).

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information processing & management*, *24*(5), 513-523.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Turian, J., Ratinov, L., & Bengio, Y. (2010). Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics* (pp. 384-394).

Schweitzer, R., Zhang, N., & Ebrahimi, M. (2018). Hacker Web Forum Collection: Hackhound Forum Dataset. University of Arizona Artificial Intelligence Lab, AZSecure-data, Director Hsinchun Chen. Available from: https://www.azsecure-data.org/other-forums.html. [Accessed: 20th February 2020]

Hackterms. (2018). A crowdsourced dictionary of coding terms. Available from: https://www.hackterms.com/about/all. [Accessed: 20th April 2020]

Techopedia. (2020). IT terms tagged with 'Hacking'. Available from https://www.techopedia.com/dictionary/tags/hacking. [Accessed: 20th April 2020]

Davies, D. L., & Bouldin, D. W. (1979). A cluster separation measure. *IEEE transactions on pattern analysis and machine intelligence*, (2), 224-227.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, *20*, 53-65.

Sood, A. K., & Enbody, R. J. (2013). Crimeware-as-a-service—a survey of commoditized crimeware in the underground market. *International Journal of Critical Infrastructure Protection*, *6*(1), 28-38.

Noroozian, A., Korczyński, M., Gañan, C. H., Makita, D., Yoshioka, K., & Van Eeten, M. (2016). Who gets the boot? analyzing victimization by ddos-as-a-service. In *International Symposium on Research in Attacks, Intrusions, and Defenses* (pp. 368-389). Springer, Cham.