2018 IEEE International Conference on Big Knowledge

KADetector: Automatic Identification of Key Actors in Online Hack Forums Based on Structured Heterogeneous Information Network

Yiming Zhang, Yujie Fan, Yanfang Ye ✉
Department of Computer Science
and Electrical Engineering
West Virginia University, WV, USA
yanfang.ye@mail.wvu.edu

Liang Zhao
Department of Information Science
and Technology
George Mason University, VA, USA
lzhao9@gmu.edu

Jiabin Wang, Qi Xiong, Fudong Shao
Tencent Anti-fraud Lab
Tencent Security Lab
Tencent, Guangdong, China
luciferwang@tencent.com

***Abstract*—Underground forums have been widely used by cybercriminals to exchange knowledge and trade in illicit products or services, which have played a central role in the cybercriminal ecosystem. In order to facilitate the deployment of effective countermeasures, in this paper, we propose and develop an intelligent system named *KADetector* to automate the analysis of Hack Forums for the identification of its key actors who play the vital role in the value chain. In *KADetector*, to identify whether the given users are key actors, we not only analyze their posted threads, but also utilize various kinds of relations among users, threads, replies, comments, sections and topics. To model the rich semantic relationships, we first introduce a structured heterogeneous information network (HIN) for representation and then use a meta-path based approach to incorporate higher-level semantics to build up relatedness over users in Hack Forums. To reduce the high computation and space cost, given different meta-paths built from the HIN, we propose a new HIN embedding model named *ActorHin2Vec* to learn the low-dimensional representations for the nodes in HIN. After that, a classifier is built for key actor identification. To the best of our knowledge, this is the first work to use structured HIN for underground participant analysis. Comprehensive experiments on the data collections from Hack Forums are conducted to validate the effectiveness of our developed system *KADetector* in key actor identification by comparisons with alternative methods.**

***Index Terms*—Heterogeneous Information Network, Network Embedding, Key Actor Identification, Online Hack Forums.**

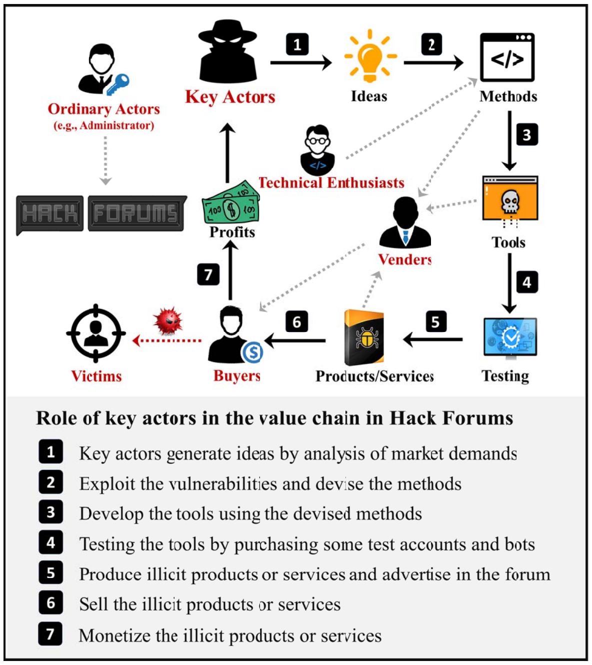
I. Introduction

Internet has become one of the most important drivers in the global economy (e.g., worldwide e-commerce sales reached over $2.14 trillions in 2017 [5]), meanwhile it also provides an open and shared platform by dissolving the barriers so that everyone has opportunity to realize his/her innovations, which implies higher prospects for illicit profits at lower degrees of risk. That is, the Internet can virtually provide a natural and excellent platform for illegal Internet-based activities, commonly known as cybercrimes [18]. Cybercrimes have become increasingly dependent on the online underground markets emerging in the forms of underground forums, through which cybercriminals can exchange knowledge (including ideas, methods and tools) and trade in illicit products (e.g., malware [11], [34], [36], stolen credit cards) or services (e.g., hacking services [8], [35]). The emerging underground forums, such as Hack Forums [2], Black Hat World [1] and Nulled [3], have enabled cybercriminals to realize considerable profits. For example, the estimated annual revenue for an individual credit card steal organization was $300 millions [21].

Underground forums have played a central role in the cybercriminal ecosystem [32], as they provide the platforms for cybercriminals to exchange knowledge and trade in the illicit products or services that facilitate all stages of cybercrimes. Using Hack Forums (one of the most prevalent underground forums) as a showcase, we investigate the profit model and monetization process. As shown in Fig. 1, the participants in the value chain can be categorized into different groups according to the roles they play: (1) ***Key actors***: This group of users play a vital role in the value chain, as they are the "decathlon" who are capable of exploiting and disseminating vulnerabilities, developing and testing malicious tools, selling and monetizing illicit products or services. For example, using Fig. 1 to further illustrate, we found that a Hack Forums user "Ban***s" (we here anonymize his user name) first ① analyzed the market demand of social media account hacking services, then ② - ③ devised a method to develop cracking tools exploiting a cookie vulnerability to perform brute-force attacks on social media accounts, later ④ purchased some compromised Facebook accounts to test his developed tools, and thus ⑤ - ⑦ sold and monetized his hacking services. We also found that "Ban***s" shared his developed cracking tools with other Hack Forums users for free, who could further use it for profits. As the skilled individuals like "Ban***s" are heavily relied upon by peers (e.g., novice hackers or newcomers can gain knowledge from them to cultivate their own specialties), they are considered as key actors within the communities. (2) ***Non-key actors***: Other users including vendors, buyers, victims, technical enthusiasts, administrators and moderators are categorized as non-key actors. Considering the vital role of key actors play in the value chain (i.e., their vulnerability exploiting capability, technical skills, cashing channel, and influence on others, etc), it is important to

978-1-5386-9125-0/18/$31.00 ©2018 IEEE
DOI 10.1109/ICBK.2018.00028

identify key actors in underground forums to facilitate the deployment of effective countermeasures. Toward this goal, human analysts need to continually spend a multitude of time to keep the latest statuses and variances of the activities in underground forums under observation. This calls for novel tools and methodologies to automate the identification of key actors to enable the law enforcement and security practitioners to devise effective interventions.



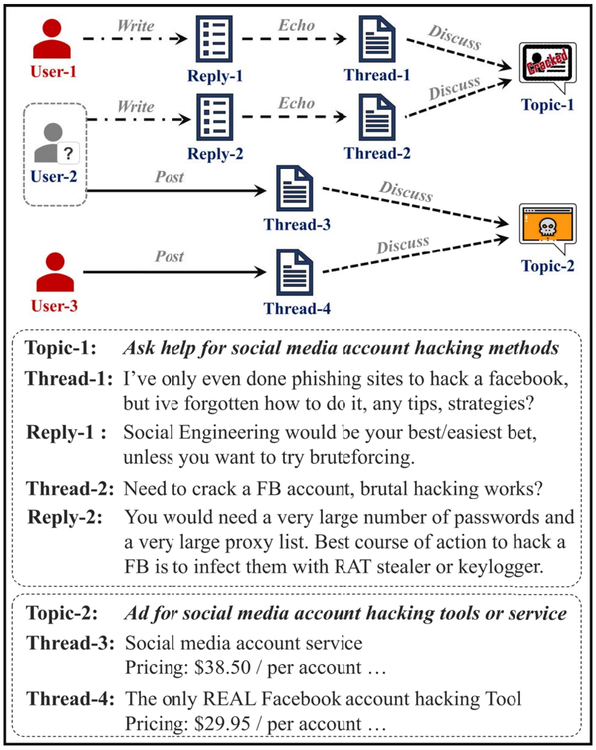
Role of key actors in the value chain in Hack Forums

1 Key actors generate ideas by analysis of market demands
2 Exploit the vulnerabilities and devise the methods
3 Develop the tools using the devised methods
4 Testing the tools by purchasing some test accounts and bots
5 Produce illicit products or services and advertise in the forum
6 Sell the illicit products or services
7 Monetize the illicit products or services

Fig. 1: Participants in the value chain in Hack Forums.

To address the above challenges, in this paper, we design and develop an intelligent system called *KADetector* to automate the analysis of Hack Forums for the identification of its key actors. In *KADetector*, to determine whether the given users are key actors, we not only analyze their posted threads, but also utilize the relations among users, threads, replies, comments, sections and topics. For example, as shown in Fig. 2, to decide whether *User-2* is a key actor, using his/her posted threads (e.g., *Thread-3*) may not be sufficient; however, with the further information that (1) *User-1* and *User-3* are key actors, (2) *User-1* and *User-2* are connected as they both wrote the replies (*Reply-1* and *Reply-2*) to answer the questions posted in the threads (*Thread-1* and *Thread-2*) indicating their expertise on social media account hacking methods (*Topic-1*), and (3) *User-2* and *User-3* are related as they both posted threads (*Thread-3* and *Thread-4*) which advertised the same kind of illicit products or service they want to sell (*Topic-2*), it can facilitate the determination if *User-2* is a key actor.

To capture such complex relationships, we first introduce a structured heterogeneous information network (HIN) [28] for representation and then use a meta-path based approach [29] to incorporate higher-level semantics to build up relatedness over users in Hack Forums. To reduce the high computation and space cost, given different meta-paths built from the HIN, we leverage network embedding techniques and propose a new HIN embedding model named *ActorHin2Vec* to learn the low-dimensional representations for the nodes in HIN. After that, a classifier is built for key actor identification. Integrating our proposed method, a system called *KADetector* is developed for key actor identification in Hack Forums which has the following major traits:



Topic-1: *Ask help for social media account hacking methods*
Thread-1: I've only even done phishing sites to hack a facebook, but ive forgotten how to do it, any tips, strategies?
Reply-1 : Social Engineering would be your best/easiest bet, unless you want to try bruteforcing.
Thread-2: Need to crack a FB account, brutal hacking works?
Reply-2: You would need a very large number of passwords and a very large proxy list. Best course of action to hack a FB is to infect them with RAT stealer or keylogger.

Topic-2: *Ad for social media account hacking tools or service*
Thread-3: Social media account service
Pricing: $38.50 / per account …
Thread-4: The only REAL Facebook account hacking Tool
Pricing: $29.95 / per account …

Fig. 2: Example of relatedness over users in Hack Forums.

- ***Novel feature representation for underground forum users:*** Instead of only using users' posted threads, we further analyze various kinds of relationships to represent underground forum users. To model different kinds of entities (i.e., user, thread, reply, comment, section and topic) and the rich semantic relationships among them, a structured HIN is first introduced to represent the Hack Forums users; and then a meta-path based approach is presented to characterize the relatedness over users.
- ***A new HIN embedding model to learn its latent representations:*** Based on the different meta-paths built from the HIN, a new HIN embedding model *ActorHin2Vec* is proposed to learn the low-dimensional representations for the nodes in HIN. Though it's proposed for key actor identification in underground forums, the HIN embedding model *ActorHin2Vec* is a general framework which is able to learn desirable node representations in HIN and thus can be further applied to various network mining tasks.
- ***An intelligent framework to automate the identification of key actors in underground forums:*** Based on the collected

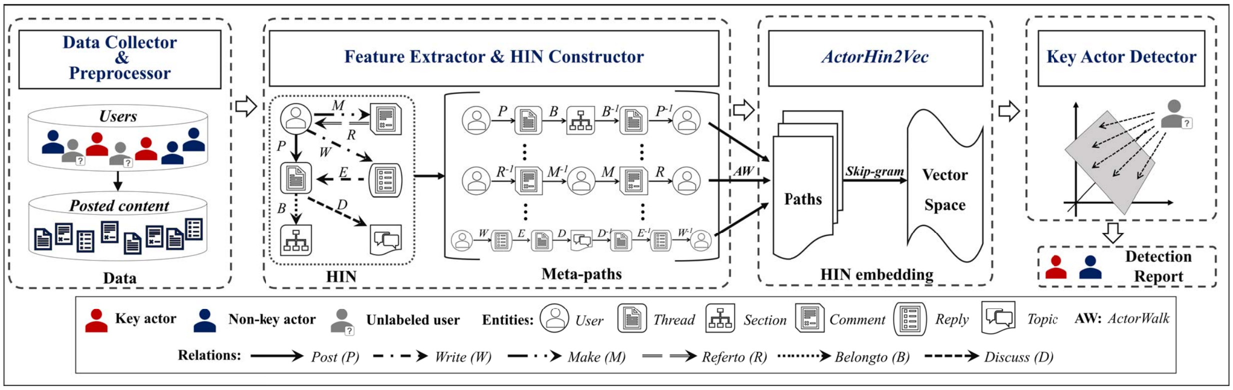


Fig. 3: System architecture of *KADetector*.

and annotated data from Hack Forums, we design and develop an intelligent system named *KADetector* to identify key actors in Hack Forums. Though we use Hack Forums as a showcase, the proposed method and developed system can be easily expanded to other promising avenues.

II. System Architecture

The system architecture of *KADetector* is shown in Fig. 3, which consists of the following components:

- ***Data Collector and Preprocessor.*** The web crawling tools are developed to collect the data from Hack Forums, including users' profiles, posted threads, replies, comments and sections, where the information of individual user is kept anonymous. For the collected threads, the preprocessor will further remove all the punctuations and stopwords, and then conduct lemmatization by using Stanford CoreNLP [22].
- ***Feature Extractor.*** Based on the data collected and preprocessed from the previous module, it first extracts the topics from each posted thread using Latent Dirichlet allocation (LDA) [6]. To depict Hack Forums users, various kinds of relations are further analyzed, including (1) *user-post-thread*, (2) *user-write-reply*, (3) *user-make-comment*, (4) *thread-belongto-section*, (5) *thread-discuss-topic*, (6) *comment-referto-user*, and (7) *reply-echo-thread*. (See Section III-A for details.)
- ***HIN Constructor.*** In this module, based on the features extracted from the previous component, a structured HIN is first presented to model the relationships among different types of entities; and then different meta-paths are built from the HIN to capture the relatedness over Hack Forums users from different views. (See Section III-B for details.)
- ***ActorHin2Vec.*** Given different meta-paths built from the HIN, a new HIN embedding model *ActorHin2Vec* is proposed to learn the low-dimensional representations for the nodes in HIN. In *ActorHin2Vec*, a new random walk method *ActorWalk* is proposed to generate the walk paths guided by different meta-path schemes, and then skip-gram is utilized to learn effective node representation for a HIN. (See Section III-C for details.)
- ***Key Actor Identifier.*** After the HIN representation learning using *ActorHin2Vec*, the mapped low-dimensional vectors of Hack Forums users will be fed to a Support Vector Machine (SVM) to train the classification model, based on which the unlabeled users can be predicted as key actors or not.

III. Proposed Method

In this section, we introduce the detailed approaches of how we represent Hack Forums users utilizing rich relationships among them, and how we solve the problem of key actor identification based on the representation.

A. Feature Extraction

As the discussion above, to depict Hack Forums users, we not only utilize their posted threads, but also consider various kinds of relationships which include the followings.

R1: To describe the relation between a user and his/her posted thread, we generate the ***user-post-thread*** matrix **P** where each element $p_{i,j} \in \{0, 1\}$ denotes if user *i* posts thread *j*.

R2: To denote the relation that a user writes a reply, we build the ***user-write-reply*** matrix **W** where each element $w_{i,j} \in \{0, 1\}$ indicates whether user *i* writes reply *j*.

R3: To represent whether a user makes a comment, we generate the ***user-make-comment*** matrix **M** where each element $m_{i,j} \in \{0, 1\}$ indicates whether user *i* makes comment *j*.

R4: To depict whether a thread belongs to a section (i.e., there are multiple sections included in Hack Forums, such as "Hacking Tools and Programs", "Monetizing Techniques", "Premium Sellers Section", etc.), we generate the ***thread-belongto-section*** matrix **B** where each element $b_{i,j} \in \{0, 1\}$ denotes if thread *i* belongs to section *j*.

R5: To represent the relation that a thread discusses a specific topic, we generate the ***thread-discuss-topic*** matrix **D** where each element $d_{i,j} \in \{0, 1\}$ indicates whether thread *i* discusses topic *j*. In our case, we use Latent Dirichlet allocation (LDA) [6] for the topic extraction from the posted threads.

R6: To denote the relation that a comment refers to (i.e., is made to) a user, we generate the ***comment-referto-user*** matrix **R** where each element $r_{i,j} \in \{0, 1\}$ indicates whether comment *i* refers to user *j*.

R7: To describe whether a reply echoes (i.e., responds to) a thread, we build the ***reply-echo-thread*** matrix **E** where element $e_{i,j} \in \{0, 1\}$ denotes if reply *i* echoes thread *j*.

B. HIN Construction

In order to depict the Hack Forums users by using the features extracted above (i.e., ***R1-R7***), we introduce how to use HIN for representation, which is capable to be composed of different types of entities and relations. We first present the concepts related to HIN as follows.

Definition 1: ***Heterogeneous information network (HIN)*** [28]. A HIN is defined as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with an entity type mapping ϕ: $\mathcal{V} \rightarrow \mathcal{A}$ and a relation type mapping ψ: $\mathcal{E} \rightarrow \mathcal{R}$, where $\mathcal{V}$ denotes the entity set and $\mathcal{E}$ is the relation set, $\mathcal{A}$ denotes the entity type set and $\mathcal{R}$ is the relation type set, and the number of entity types $|\mathcal{A}| > 1$ or the number of relation types $|\mathcal{R}| > 1$. The ***network schema*** [28] for a HIN $\mathcal{G}$, denoted as $\mathcal{T}_{\mathcal{G}} = (\mathcal{A}, \mathcal{R})$, is a graph with nodes as entity types from $\mathcal{A}$ and edges as relation types from $\mathcal{R}$.

In our case, i.e., the identification of key actors in Hack Forums, we have six entity types (i.e., user, thread, reply, comment, section, and topic) and seven types of relations among them (i.e., ***R1-R7***). Based on the definitions above, the network schema for HIN in our application is shown in Fig. 4, which enables the Hack Forums users to be represented in a comprehensive way that utilizes both their posted content and relation-based information simultaneously.

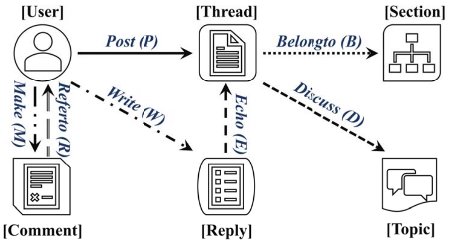


Fig. 4: Network schema for HIN.

To enrich the semantics of relatedness among Hacker Forum users, the concept of meta-path has been proposed [29] to formulate the higher-order relationships among entities in HIN. Here, we follow this concept and extend it to our application of key actor identification in Hack Forums.

Definition 2: ***Meta-path*** [29]. A meta-path $\mathcal{P}$ is a path defined on the graph of network schema $\mathcal{T}_{\mathcal{G}} = (\mathcal{A}, \mathcal{R})$, and is denoted in the form of $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} ... \xrightarrow{R_L} A_{L+1}$, which defines a composite relation $R = R_1 \cdot R_2 \cdot \ldots \cdot R_L$ between types A_1 and A_{L+1}, where $\cdot$ denotes relation composition operator, and L is the length of $\mathcal{P}$.

In our application, based on the seven different kinds of relationships, we design eight meaningful meta-paths for characterizing relatedness over Hack Forums users, i.e., ***PID1-PID8*** shown in Fig. 5. Different meta-paths depict the relatedness between two users at different views. For example, a typical one is ***PID1***: $user \xrightarrow{post} thread \xrightarrow{belongto} section \xrightarrow{belongto^{-1}} thread \xrightarrow{post^{-1}} user$ which means that two users can be connected through the path that their posted threads belong to the same section; while another meta-path ***PID2***: $user \xrightarrow{post} thread \xrightarrow{discuss} topic \xrightarrow{discuss^{-1}} thread \xrightarrow{post^{-1}} user$ denotes that two users can be connected if their posted threads discuss the same topic. In our application, meta-path is a straightforward method to connect users via different relationships among different entities in HIN, and enables us to portray the relatedness over Hack Forums users in a comprehensive way.

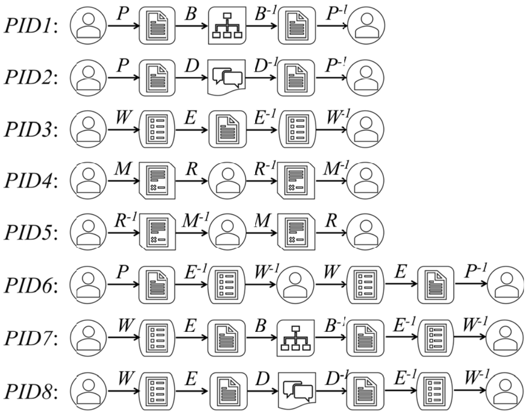


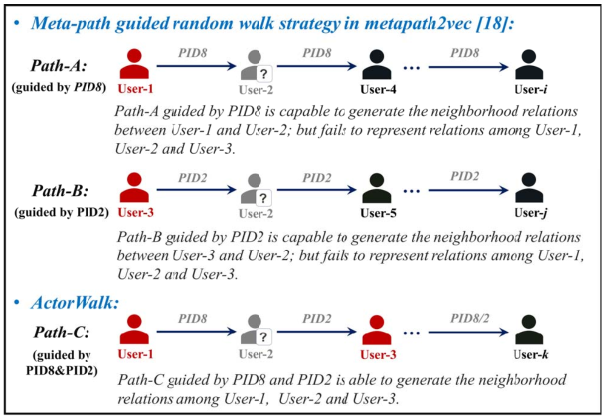
Fig. 5: Meta-paths built for key actor identification. (The symbols in the figure are the abbreviations shown in Fig. 4.)

C. ActorHin2Vec

In order to reduce the high computation and space cost for network mining, scalable representation learning method for HIN is in need. We first formalize the problem of HIN representation learning as below.

Definition 3: ***HIN representation learning*** [9]. Given a HIN $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the representation learning task is to learn a function $f : \mathcal{V} \rightarrow \mathbb{R}^d$ that maps each node $v \in \mathcal{V}$ to a vector in a d-dimensional space $\mathbb{R}^d$, $d \ll |\mathcal{V}|$ that are capable to preserve the structural and semantic relations among them.

To solve the problem of HIN representation learning, due to the heterogeneous property of HIN, it is difficult to directly apply the homogeneous network embedding techniques (e.g., DeepWalk [26], LINE [30], node2vec [14]) to learn the latent representations for HIN. To address this issue, HIN embedding methods such as metapath2vec [9] was proposed. In metapath2vec, given a meta-path scheme, it employs meta-path based random walk and heterogeneous skip-gram to learn the latent representations for HIN. It was proposed to support one meta-path scheme to guide the walker traversing HIN; however, in our application, Hack Forums users can be connected through eight different meta-path schemes. It may not be feasible to directly employ metapath2vec in our case for key actor identification. To put this into perspective, as the example described in Section I illustrated by Fig. 2, if we directly apply the meta-path guided random walk strategy in metapath2vec, as shown in Fig. 6, *User-2* can either be connected to *User-1* (i.e., *Path-A* guided by meta-path scheme *PID8*) or *User-3* (*Path-B* guided by meta-path scheme *PID2*); in other words, it fails to generate the neighborhood relations among *User-1*, *User-2* and *User-3*. In fact, such neighborhood relationships can be generated based on the path via random walks guided by a combination of meta-path schemes of *PID8* and *PID2* (i.e., *Path-C* guided by *PID8* and *PID2*). To address this issue, we design a new HIN embedding model *ActorHin2Vec* to learn node representations in HIN: first, a new random walk method *ActorWalk* is proposed to generate walk paths guided by different meta-path schemes; then skip-gram is utilized to learn node representation for a HIN.



• Meta-path guided random walk strategy in metapath2vec [18]:
Path-A:
(guided by PID8)
PID8
PID8
PID8
User-1
User-2
?
User-4
User-i
Path-A guided by PID8 is capable to generate the neighborhood relations between User-1 and User-2; but fails to represent relations among User-1, User-2 and User-3.
Path-B:
(guided by PID2)
PID2
PID2
PID2
User-3
User-2
?
User-5
User-j
Path-B guided by PID2 is capable to generate the neighborhood relations between User-3 and User-2; but fails to represent relations among User-1, User-2 and User-3.
• ActorWalk:
Path-C:
(guided by PID8&PID2)
PID8
PID2
PID8/2
User-1
User-2
?
User-3
User-k
Path-C guided by PID8 and PID2 is able to generate the neighborhood relations among User-1, User-2 and User-3.

Fig. 6: Single meta-path guided random walk vs. multiple meta-paths guided random walk strategy (i.e., *ActorWalk*).

ActorWalk. Traditional random walk mechanism relies on the normalized probability distributed over the neighbors of current node by ignoring their node types, which is unable to capture the semantic and structural correlations among different types of nodes in a HIN. Here, we show how to use different meta-path schemes to guide random walker in a HIN to generate the paths of multiple types of nodes. In our application, given a HIN $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, at each step, assume that the current node is u and the next node is v with entity type of γ, the transition probability from u to v is defined as:

$$p(v|u) = \frac{1}{|N_\gamma(u)|}, \tag{1}$$

where $N_\gamma(u)$ denotes γ type of neighborhood of node u (if $N_\gamma(u) = 0$, then we set $p(v|u) = 0$). Given a set of different meta-path schemes, *ActorWalk* starts from the source node u, and then employs the proposed meta-path scheme selection strategy to guide the walks for path generation. We here used an example to illustrate the meta-path scheme selection strategy in *ActorWalk*. To simplify the representation, we denote a meta-path $\mathcal{P}$ as $A_1 A_2 \cdots A_{L+1}$. Given a set of different meta-path schemes (i.e., *PID1-PID3*) denoted as **P** = $\{$*UTSTU*,*UTtTU*,*URTRU*$\}$ (*U*: entity type of *User*, *T*: *Thread*, *S*: *Section*, *t*: *topic* and *R*: *Reply*), the *ActorWalk* starts from source node u of type *U* and the set of candidate meta-path schemes being selected to guide the walks *C* is initialized as **P**. At each step, to decide the next node to walk, it first generates the candidate type set **S** for the next node (i.e., **S** = $\{$*T*,*R*$\}$ in this example) and then randomly chooses a type from **S** (assume *T* is selected here); later, it draws a neighborhood v of type *T* by Eq.(1) and adds it to the path; afterwards, it updates *C* = $\{$*UTSTU*,*UTtTU*$\}$ (i.e., by removing *PID3*). Each iteration stops when $C = \emptyset$. *ActorWalk* repeats the iterations to generate a walk path until the length of the path achieves the pre-defined walk length l. The pseudo code of the proposed *ActorWalk* is shown in Algorithm 1. Compared with the metapath2vec [9], *ActorWalk* can generate the walk paths guided by a set of different meta-paths built from HIN, which has more expressive semantics. Note that, the metapath2vec is a special case of *ActorWalk* (i.e., if the input meta-path scheme set only includes one single meta-path).

Algorithm 1: The *ActorWalk* algorithm.

```
Input: HIN $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a set of meta-path schemes
       $\{\mathcal{P}_k\}_{k=1}^m$, source node $u$, walk length $l$
Output: random walk path $W$
initialize walk path $W = [u]$, the set of candidate
  meta-path schemes $C = \{\mathcal{P}_k\}_{k=1}^m$;
for $i = 2$ to $l$ do
    if $C = \emptyset$ then
        $C = \{\mathcal{P}\}_{k=1}^m$;
    end
    generate the candidate type set $\mathbf{S}$ for the next node;
    if $\mathbf{S} = \emptyset$ then
        $C = \emptyset$;
        continue;
    else
        randomly choose a type $\gamma$ from $\mathbf{S}$;
    end
    draw neighborhood $v$ of type $\gamma$ by Eq. (1);
    if $v$ doesn't exist then
        break;
    else
        append $v$ to $W$;
    end
    update $C$;
    if $\exists \mathcal{P}_k$ is the sub-path of $\mathcal{P}_j$ ($\mathcal{P}_j \in C - \mathcal{P}_k$) and
      $j \neq k$ then
        randomly choose a $\mathcal{P}_k$ from $C$ as $C$;
    end
end
```

Skip-gram. After sampling paths in HIN via *ActorWalk*, skip-gram [23] is then applied on the paths to maximize the probability of observing a node's neighbourhood (within a window w) conditioned on its current representation. The objective function of skip-gram is:

$$\arg\max_{Y} \sum_{-w \le k \le w, j \ne k} \log p(v_{j+k} | Y(v_j)), \tag{2}$$

where $Y(v_j)$ is the current representation vector of v_j, $p(v_{j+k}|Y(v_j))$ is defined using the softmax function:

$$p(v_{j+k}|Y(v_j)) = \frac{\exp(Y(v_{j+k}) \cdot Y(v_j))}{\sum_{q=1}^{|V|} \exp(Y(v_q) \cdot Y(v_j))}. \tag{3}$$

Due to its efficiency, we first apply hierarchical softmax technique [24] to solve Eq. 3; then the stochastic gradient descent (SGD) [7] is used to train the skip-gram.

TABLE I: Comparisons of *ActorHin2Vec* with other network representation learning methods in key actor identification.

Metric	Method	10%	20%	30%	40%	50%	60%	70%	80%	90%
ACC	DeepWalk	0.637	0.675	0.736	0.743	0.759	0.762	0.782	0.791	0.796
	LINE	0.673	0.733	0.789	0.802	0.815	0.828	0.832	0.837	0.839
	metapath2vec	0.714	0.755	0.809	0.817	0.840	0.844	0.848	0.847	0.854
	ActorHin2Vec	0.749	0.781	0.816	0.830	0.844	0.853	0.865	0.870	0.873
F1	DeepWalk	0.450	0.519	0.561	0.585	0.569	0.587	0.629	0.635	0.641
	LINE	0.489	0.561	0.643	0.659	0.676	0.679	0.696	0.705	0.712
	metapath2vec	0.551	0.569	0.665	0.687	0.705	0.709	0.721	0.725	0.735
	ActorHin2Vec	0.580	0.621	0.678	0.696	0.715	0.733	0.753	0.761	0.773

IV. Experimental Results And Analysis

In this section, we fully evaluate the performance of our developed system *KADetector* for key actors identification by comparisons with other alternate methods using the data collection from Hack Forums [2].

A. Data Collection and Annotation

We develop a set of crawling tools to collect the users' profiles, posted threads, replies and comments as well as the sections in Hack Forums in a period of time. By the date, we have collected ***3,602,431 threads*** posted by ***251,186 users***, through September 2007 to May 2018. To obtain the ground truth, we randomly select 4,000 users who have posted threads in the sub-forum of "Hack" and in the section of "Premium Sellers Section". Then, we have spent 3 months to label whether the Hack Forums users are key actors or not following the criteria of: (1) the key actors should be active in the forum (e.g., post rate per day is greater than 1 and time spent online per day is greater than 0.5 hours); and (2) they should be capable of exploiting and disseminating vulnerabilities, developing and testing malicious tools, selling and monetizing illicit products or services (e.g., reputation is greater than 500, number of total reviews for sold products or service is great than 200, each product etc). The mutual agreement is above 95%, and only the ones with agreements are retained. Based on these criteria, ***892*** are labeled as *key actors* and ***2,940*** are *non-key actors*. We use the accuracy (ACC) and F1 measure to evaluate the effectiveness of different methods in key actor identification in Hack Forums.

B. Evaluation of ActorHin2Vec

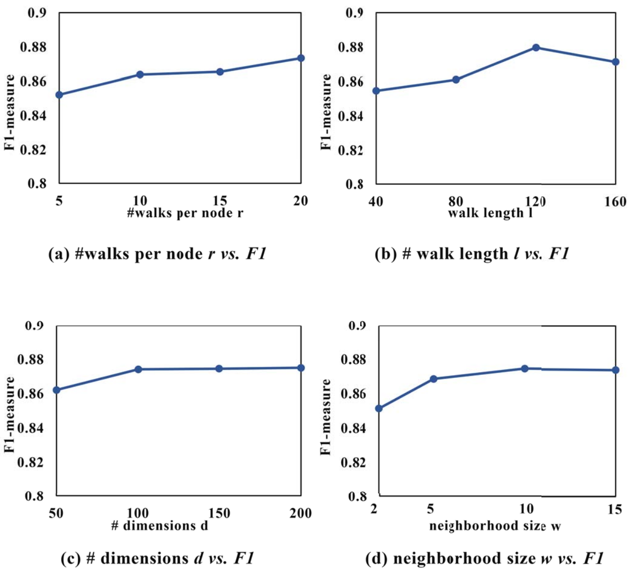
In this set of experiments, we evaluate our proposed method *ActorHin2Vec* by comparisons with several recent network representation learning methods: DeepWalk [26], LINE [30] and metapath2vec [9]. For DeepWalk and LINE, we ignore the heterogeneous property of HIN and directly feed the HIN for representation learning; in metapath2vec, a walk path will be generated based on a single meta-path scheme. The parameter settings used for *ActorHin2Vec* are in line with typical values used for DeepWalk, LINE and metapath2vec: vector dimension $d = 128$ (LINE: 64 for each order (1st- and 2nd-order)), walks per node $r = 10$, walk length $l = 80$ and window size $w = 10$. To facilitate the comparisons, we use the experimental procedure as in [9], [26], [30]: we randomly select a portion of training data described in Section IV-A (ranging from 10% to 90%) for training and the remaining ones for testing. The SVM is used as the classification model for all the methods where the penalty is empirically set to be 10 while other parameters are set by default. Table I illustrates the results of different network representation learning methods in key actor identification. From Table I, we can see that the proposed *ActorHin2Vec* model consistently and significantly outperforms all baselines for key actor identification in terms of *ACC* and *F1*. That is to say, *ActorHin2Vec* learns significantly better user representations than current state-of-the-art methods. The success of *ActorHin2Vec* lies in the proper consideration and accommodation of the heterogeneous property of HIN (i.e., the multiple types of nodes and relations), and the advantage of the proposed *ActorWalk* for sampling the node paths.

C. Comparisons with Traditional Machine Learning Methods

In this set of experiments, we compare *KADetector* with other traditional machine learning methods. For these methods, we construct three types of features: ***f-1***: content-based features (i.e., each user's posted thread(s) represented by a bag-of-words vector); ***f-2***: four relation-based features associated with Hack Forums users (i.e., whether two users i) belong to the same section, ii) reply to the same thread, iii) commend on the same user; and iv) are made comments by the same user); ***f-3***: augmented features of *f-1* and *f-2*. Based on these features, we consider two typical classification models, i.e., Naive Bayes (NB) and SVM. The experimental results are illustrated in Table II. From the results we can observe that feature engineering (***f-3***: concatenation of different features altogether) helps the performance of machine learning, but *KADetector* added the knowledge represented as HIN significantly outperforms other baselines. This again demonstrates that, to identify the key actor in underground forums, *KADetector* using meta-path based approach over HIN is able to build the higher-level semantic and structural connection between users with a more expressive and comprehensive view and thus achieves better identification performance.

TABLE II: Comparisons of other machine learning methods.

Method	NB			SVM			*KADetector*
Settings	***f-1***	***f-2***	***f-3***	***f-1***	***f-2***	***f-3***	/
ACC	0.692	0.675	0.727	0.723	0.709	0.771	0.880
F1	0.515	0.493	0.559	0.552	0.524	0.622	0.775



(a) #walks per node *r vs. F1*

(b) # walk length *l vs. F1*

(c) # dimensions *d vs. F1*

(d) neighborhood size *w vs. F1*

Fig. 7: Parameter sensitivity evaluation.

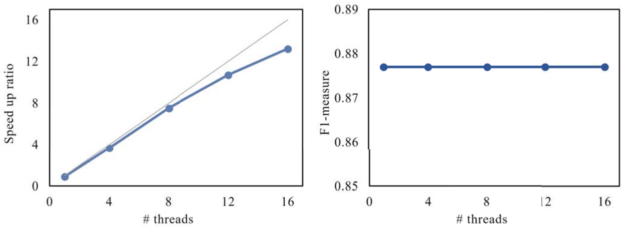


Fig. 8: *Left*: Speed-up vs # threads, *Right*: F1 vs # threads.

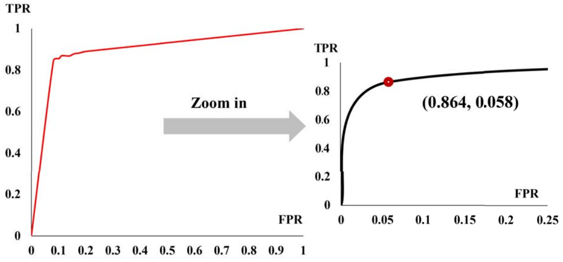


Fig. 9: ROC curve.

D. Parameter Sensitivity, Scalability and Stability

In this set of experiments, we first conduct the **sensitivity** analysis of how different choices of parameters will affect the performance of *ActorHin2Vec* in key actor identification. From the results shown in Fig. 7(a) and 7(b), we can observe that the balance between computational cost (number of walks per node r and walk length l in x-axis) and efficacy (F1 in y-axis) can be achieved when r = 20 and l = 120. We also examine how latent dimensions (d) and neighborhood size (w) affect the performance. As shown in Fig. 7(c) and Fig. 7(d), we can see that the performance tends to be stable once d reaches around 100 or when w increases to 10. Overall, *ActorHin2Vec* is not strictly sensitive to these parameters and is able to reach high performance under a cost-effective parameter choice. We then further evaluate the **scalability** of *ActorHin2Vec* which can be parallelized for optimization. We run the experiments using the default parameters with different number of threads (i.e., 1, 4, 8, 12, 16), each of which utilizes one CPU core. Fig. 8.*left* shows the speed-up of *ActorHin2Vec* deploys multiple threads over the single-threaded case, which shows that the model achieves acceptable sub-linear speed-ups as the line is close to the optimal line; while Fig. 8.*right* shows that the performance remains stable when using multiple threads for model updating. Overall, the proposed system are efficient and scalable for large-scale HIN with large numbers of nodes. For **stability** evaluation, Fig. 9 shows the overall receiver operating characteristic (ROC) curves of *KADetector* based on the ten-fold cross validations; it achieves an impressive 0.864 average TP rate at the 0.058 average FP rate for key actor identification.

V. Related Work

To combat the cybercrimes which have become increasingly dependent on the online underground forums, there have been many research efforts on underground forum analysis [4], [17], [20], [25], which can be categorized into three areas [16]: (1) the first type focuses on identifying the threats found in the content and other content-related features; (2) the second type of research mainly works on understanding the cybercriminal community structure and social relationships; (3) the third type focuses on identifying the most influential cybercriminal community members. Our work is one of the third type. There have been some existing works on key member identification, for examples: Yang et al. [33] incorporated the message similarity and response immediacy features with link analysis to determine the impact and the neighborhood of the influential users; Tang et al. [31] used Bipartite Graph analysis and developed a user interest and topic detection model to predict user participation in the Dark Web; other advanced techniques including deep learning [19] are developed to profile sellers from their advertisements. In this paper, we propose to utilize not only the posted threads, but also various kinds of relationships to represent underground forum users for key actor identification. Based on the extracted features, the users are represented by a structured HIN.

HIN is used to model different types of entities and relations, and has been applied to various applications, such as scientific publication network analysis [27], [29], biomedical knowledge mining [12], [13] and malware detection [10], [15]. Several measures (e.g., meta-path [29]) have already been proposed for relevance computation over HIN entities. To reduce the high computation and space cost in network mining, many efficient network embedding methods have been proposed to address representation learning for homogeneous network, such as DeepWalk [26], node2vec [14], and LINE [30]. Unfortunately, due to the heterogeneous properties of HIN, it's difficult to directly apply them for HIN representation learning. To tackle this challenge, metapath2vec [9] have been proposed for HIN representation learning. However, metapath2vec can only support one meta-path scheme to guide the walker traversing HIN. In our application, the relatedness of Hack Forums users are depicted by different meta-paths; therefore, it calls for new efficient HIN embedding methods. To address this issue, given different meta-path schemes built from the HIN, we propose a new HIN embedding model *ActorHin2Vec* to learn the desirable node representations in HIN which is capable to preserve both the semantics and structural correlations between different types of nodes in HIN.

VI. Conclusion

To combat the illicit activities in underground forums, in this paper, we design and develop an intelligent system named *KADetector* to automate the identification of key actors in Hack Forums. In *KADetector*, we first construct a structured HIN to leverage users' posted threads and the rich relationships among users, threads, replies, comments, sections and topics, which gives the user a higher-level semantic representation. This is the first attempt to use HIN to depict underground forum users. Then, a meta-path based approach is used to characterize the semantic relatedness over Hack Forums users. To learn the low-dimensional representations for the nodes in HIN, based on different meta-path schemes built from the HIN, a new HIN embedding model *ActorHin2Vec* is proposed. After that, a classifier is built for key actor identification in Hack Forums. The promising experimental results on the collected and annotated data sets from Hack Forums demonstrate that *KADetector* integrated our propose method outperforms alternative approaches.

Acknowledgment

This work is partially supported by the U.S. National Science Foundation under grants OAC-1839909, CNS-1618629 and CNS-1814825, NIJ 2018-75-CX-0032, WV Higher Education Policy Commission Grant (HEPC.dsr.18.5), and WVU Research and Scholarship Advancement Grant (R-844).

References

[1] *Black Hat world*, https://www.blackhatworld.com/.

[2] *Hacker Forums*, https://hackforums.net/.

[3] *Nulled*, https://www.nulled.to.

[4] V. Benjamin and H. Chen, "Securing cyberspace: Identifying key actors in hacker communities," in *ISI*, 2012, pp. 24–29.

[5] K. T. Blagoeva and M. Mijoska, "Applying tam to study online shopping adoption among youth in the republic of macedonia," in *Management International Conference*, 2017.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *JMLR*, vol. 3, no. Jan, pp. 993–1022, 2003.

[7] L. Bottou, "Stochastic gradient learning in neural networks," *Proceedings of Neuro-Nımes*, vol. 91, no. EC2, 1991.

[8] L. Chen, S. Hou, and Y. Ye, "Securedroid: Enhancing security of machine learning-based detection against adversarial android malware attacks," in *ACSAC*. ACM, 2017, pp. 362–372.

[9] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *KDD*. ACM, 2017, pp. 135–144.

[10] Y. Fan, S. Hou, Y. Zhang, Y. Ye, and M. Abdulhayoglu, "Gotcha-sly malware!: Scorpion a metagraph2vec based malware detection system," in *KDD*. ACM, 2018, pp. 253–262.

[11] Y. Fan, Y. Ye, and L. Chen, "Malicious sequential pattern mining for automatic malware detection," *Expert Systems with Applications*, vol. 52, pp. 16–25, 2016.

[12] Y. Fan, Y. Zhang, Y. Ye, and X. Li, "Automatic opioid user detection from twitter: Transductive ensemble built on different meta-graph based similarities over heterogeneous information network." in *IJCAI*, 2018, pp. 3357–3363.

[13] Y. Fan, Y. Zhang, Y. Ye, W. Zheng *et al.*, "Social media for opioid addiction epidemiology: Automatic detection of opioid addicts from twitter and case studies," in *CIKM*. ACM, 2017, pp. 1259–1267.

[14] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *KDD*. ACM, 2016, pp. 855–864.

[15] S. Hou, Y. Ye, Y. Song, and M. Abdulhayoglu, "Hindroid: An intelligent android malware detection system based on structured heterogeneous information network," in *KDD*. ACM, 2017, pp. 1507–1515.

[16] S.-Y. Huang and H. Chen, "Exploring the online underground marketplaces through topic-based social network and clustering," in *ISI*, 2016, pp. 145–150.

[17] A. Hudic, K. Krombholz, T. Otterbein, C. Platzer, and E. Weippl, "Automated analysis of underground marketplaces," in *IFIP International Conference on Digital Forensics*. Springer, 2014, pp. 31–42.

[18] E. Kraemer-Mbula, P. Tang, and H. Rush, "The cybercrime ecosystem: Online innovation in the shadows?" *Technological Forecasting and Social Change*, vol. 80, no. 3, pp. 541–555, 2013.

[19] W. Li and H. Chen, "Identifying top sellers in underground economy using deep learning-based sentiment analysis," in *JISIC), 2014 IEEE Joint*. IEEE, 2014, pp. 64–67.

[20] W. Li, H. Chen, and J. F. Nunamaker Jr, "Identifying and profiling key sellers in cyber carding community: Azsecure text mining system," *Journal of Management Information Systems*, vol. 33, no. 4, pp. 1059–1086, 2016.

[21] G. D. Maio, A. Kapravelos, Y. Shoshitaishvili, C. Kruegel, and G. Vigna, "Pexy: The other side of exploit kits," in *DIMVA*, 2014, pp. 132–151.

[22] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit." in *ACL (System Demonstrations)*, 2014, pp. 55–60.

[23] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[24] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *NIPS*, 2013, pp. 3111–3119.

[25] M. Motoyama, D. McCoy, K. Levchenko, S. Savage, and G. M. Voelker, "An analysis of underground forums," in *SIGCOMM*. ACM, 2011, pp. 71–80.

[26] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *KDD*. ACM, 2014, pp. 701–710.

[27] Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han, "Co-author relationship prediction in heterogeneous bibliographic networks," in *ASONAM*. IEEE, 2011, pp. 121–128.

[28] Y. Sun and J. Han, "Mining heterogeneous information networks: principles and methodologies," *DMKD*, vol. 3, no. 2, pp. 1–159, 2012.

[29] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *VLDB Endowment*, vol. 4, no. 11, pp. 992–1003, 2011.

[30] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *WWW*. International World Wide Web Conferences Steering Committee, 2015, pp. 1067–1077.

[31] X. Tang, C. C. Yang, and M. Zhang, "Who will be participating next?: predicting the participation of dark web community," in *KDD Workshop on Intelligence and Security Informatics*. ACM, 2012, p. 1.

[32] K. Thomas, D. Yuxing, H. David, W. Elie, B. C. Grier, T. J. Holt, C. Kruegel, D. McCoy, S. Savage, and G. Vigna, "Framing dependencies introduced by underground commoditization," in *Proceedings of the Workshop on Economics of Information Security (WEIS)*, 2015.

[33] C. C. Yang, X. Tang, and B. M. Thuraisingham, "An analysis of user influence ranking algorithms on dark web forums," in *KDD Workshop on Intelligence and Security Informatics*. ACM, 2010, p. 10.

[34] Y. Ye, L. Chen, S. Hou, W. Hardy, and X. Li, "Deepam: a heterogeneous deep learning framework for intelligent malware detection," *Knowledge and Information Systems*, vol. 54, no. 2, pp. 265–285, 2018.

[35] Y. Ye, T. Li, D. Adjeroh, and S. S. Iyengar, "A survey on malware detection using data mining techniques," *ACM Computing Surveys (CSUR)*, vol. 50, no. 3, p. 41, 2017.

[36] Y. Ye, T. Li, S. Zhu, W. Zhuang, E. Tas, U. Gupta, and M. Abdulhayoglu, "Combining file content and file relations for cloud based malware detection," in *KDD*. ACM, 2011, pp. 222–230.