Hindawi
Security and Communication Networks
Volume 2020, Article ID 8387672, 9 pages
https://doi.org/10.1155/2020/8387672

*Research Article*

# User Audit Model Based on Attribute Measurement and Similarity Measurement

**Xiaohui Yang and Ying Sun**

*School of Cyber Security and Computer, Hebei University, Baoding 071002, China*

Correspondence should be addressed to Xiaohui Yang; yxh@hbu.edu.cn

Received 25 October 2019; Accepted 18 January 2020; Published 9 March 2020

Guest Editor: Geethapriya Thamilarasu



The Internet of Things (IoT) is an open network. And, there are a large number of malicious nodes in the network. These malicious nodes may tamper with the correct data and pass them to other nodes. The normal nodes will use the wrong data for information dissemination due to a lack of ability to verify the correctness of the messages received, resulting in the dissemination of false information on medical, social, and other networks. Auditing user attributes and behavior information to identify malicious user nodes is an important way to secure networks. In response to the user nodes audit problem, a user audit model based on attribute measurement and similarity measurement (AM-SM-UAM) is proposed. Firstly, the user attribute measurement algorithm is constructed, using a hierarchical decision model to construct a judgment matrix to analyze user attribute data. Secondly, the blog similarity measurement algorithm is constructed, evaluating the similarity of blog posts published by different users based on the improved Levenshtein distance. Finally, a user audit model based on a security degree is built, and malicious users are defined by security thresholds. Experimental results show that this model can comprehensively analyze the attribute and behavior data of users and have more accurate and stable performance in the practical application of the network platforms.

## 1. Introduction

The Internet of Things (IoT) is the latest evolution of the Internet, including a great deal of connected physical devices and applications [1]. IoT allows object collection and data exchange, etc. [2], which can perform medical data management, medical information monitoring, and user information analysis. At present, problems such as violating the privacy of medical data and publishing false medical advertisements often appear in the network, and malicious users become more and more complicated and hidden, which brings great security threats to networks. Accurate and rapid identification of malicious users not only benefits the security of the user's data and information but also facilitates timely response to threats in networks.

When objects connected to the Internet of Things continue to generate information and report to Internet users, a noteworthy development is that they will also join traditional social networks and interact with "people" in social networks. Social networks are not just person-to-person social, but person-to-person, person-to-thing, and thing-to-thing. Therefore, malicious users in social networks will inevitably pose a threat to the security of the Internet of Things.

To identify malicious users in social networks and ensure the security of the Internet of Things, a user audit model based on attribute and similarity measures is proposed. The model measures the similarity between complex user attributes and users, analyzes the user's attribute information and behavior information, determines the user's security index, and finds the similarity of self-issued behavior among users, which improves the accuracy of the model to identify malicious users. At the same time, the concept of user security is proposed to measure user security in the Internet of Things, which is an important indicator to identify malicious user nodes.

The contributions of this paper are listed as follows:

(1) Construct user attribute measurement algorithm, obtain user attribute data, calculate attribute weight vector by hierarchical weight decision model, and analyze attribute information.

(2) Construct similarity measurement algorithm, consider user blog text information, use word segmentation technology, extract original blog content keywords, and improve Levenshtein distance. By studying the contents of blog posts, it reflects the preferences and characteristics of users' spontaneous behaviors.

(3) Propose the concept of user security degree as an important distinguishing indicator between normal users and malicious users. At the same time, the security threshold is defined, security threshold judgment based on user security degree to identify malicious users.

(4) Analyze the performance of the model in the real microblog dataset and compare it with other algorithm models. AM-SM-UAM has better performance in improving the accuracy, stability, and model parameter tuning of malicious user nodes.

The rest of this paper is organized as follows. In Section 2, we provide a brief introduction to existing related work. The model is described in section 3. In Section 4, we introduce the draft model AM-SM-UAM in detail. In Section 5, we introduce the experimental results. Finally, we conclude our work in Section 6.

## 2. Related Work

In recent years, malicious user identification methods based on abnormal behavior detection have attracted considerable attention. Hajmohammadi et al. [3]used active learning to automatically obtain malicious users, which has the problems of large computational overhead, information redundancy, and information overload. Gupta et al. used feature extraction methods, such as text features [4, 5] and network structure features [6–8], to extract distinguishing features from a large number of marked normal users and malicious users to train the user classification model. Due to different evaluation criteria of the extracted distinguishing user features in diverse application backgrounds, the detection accuracy is low and the stability is poor. Lee et al. [9] attracted malicious users to actively attract attention by adding trapping nodes to the network and obtained the behavior characteristics of malicious users separate from normal users. The detection framework based on the trapping system was used to determine malicious users of MySpace and Twitter. Zhang et al. [10] and Tahir et al. [11] analyzed the effect of collaborative learning on clustering, and the accuracy of the identification of malicious users was minimal. Meng and Kwok [12] corrected the false alarm rate of abnormal intrusion detection based on SVM. Although partially labeled training samples were used to reduce the system overhead, most training samples were assumed to be uniform and average, and the actual situation is sometimes difficult to meet the condition, often overfitting phenomenon. Zhu et al. [13] proposed a social group identification method based on local attribute community detection. Owing to a large number of adjacent nodes, the computational overhead is relatively large. Abnormal behavior detection methods based on user relationship, such as Ju et al. [14], based on the calculation model of compactness centrality and credit, judged the influence of users by user relationship adjacency matrix; Li et al. [15] proposed the PageRank based on account anomaly detection algorithm, which builds a social relationship matrix based on the user relationship and ranks the account to detect malicious users through the iterative calculation of PageRank value. This method does not consider the user's attribute characteristics, and the ranking result of the user is affected by the time delay, so the accuracy rate is minimal in the IoT with an uneven scale.

In summary, existing malicious user identification methods have three important shortcomings. First, user data samples are required to be high, the test results are unstable, and the evaluation indexes such as computational efficiency and accuracy cannot be the best of both worlds. Second, feature extraction, clustering, and other methods only consider the user attribute characteristics or only consider the user relationship information, without considering the user spontaneous behavior, the detection of social user attribute information, and spontaneous behavior information. Third, only numerical characteristics are considered, and text data such as user blog information are not considered.

In the era of mobile Internet, the Internet of Things needs to store, calculate, and analyze data through the service management layer when it implements information processing functions. It uses existing or perceived information to create new information. During development, it is necessary not only to configure the device network but also to perform user system development, data processing, etc. At this time, the Internet of Things to hardware also has social attributes. Therefore, to maintain the security of the Internet of things and identify malicious users in the network, in response to the above problems, a user audit model based on attribute measurement and similarity measurement (AM-SM-UAM) is proposed by taking the social platform of microblog with a large user volume as an example. AM-SM-UAM defines the concept of user security degree and builds an attribute measurement algorithm and a similarity measurement algorithm to audit user attribute information and behavior information and to identify malicious user nodes in the microblog.

## 3. Model Description

The key to the construction of the AM-SM-UAM is to rationally quantify the user's attribute information and behavior information, to realize the identification of malicious users and to ensure the smooth operation of the microblog. A series of operations, such as analyzing users' information and measuring user attributes and the similarity of blog content, is meant by user audit.

Microblog user set, $U = \{u_i\}$ $(i = 1, \ldots\ldots, n)$, represents the collection of microblog users including malicious users and normal users, and malicious user um, um $\in U$, represents the malicious user identified by the user audit. Then, the problem of auditing microblog users to identify malicious users is defined as follows: how to perform user auditing on the user set $U$ in the microblog and determine the malicious user um by constructing the attribute measurement algorithm and similarity measurement algorithm. AM-SM-UAM consists of three layers as shown in Figure 1.

(1) Data layer: read the original data and preprocess the data. The user vector is constructed, and the valid user attribute information and user blog text information in the original data are selected.

(2) Feature layer: user attribute information and blog text information are constructed based on user features. Attribute vectors are established based on user attribute features, and user attributes are represented by numerical values. The text information of user blog is analyzed by using the word segmentation technology, the keywords are extracted to represent user blog, and the user text data are processed to achieve the purpose of simultaneously processing and analyzing both numerical data and text data.

(3) Audit layer: two targeted algorithm strategies are proposed to implement user auditing. First, an attribute measurement algorithm is constructed to quantify user attribute information. Establish a hierarchical decision model, construct a judgment matrix, and calculate the user's own attribute values. Use the hierarchical decision model to calculate the user attribute weight vector, so that the relative importance of the user's various attribute information can be clearly expressed. Second, the similarity measurement algorithm is constructed to process users' blog information and evaluate the similarity of users with different attribute values in blog keywords, so as to achieve the purpose of computing the similarity of textual data. The user's attribute information and blog text information are considered comprehensively from the two aspects of user attribute and spontaneous behavior to obtain user security degree.

## 4. Model Construction

When AM-SM-UAM audits the attribute information and behavior information of microblog users, it comprehensively considers the user attribute features and blog content information and measures the user's security degree by measuring the user's attributes and calculating the similarity between user blogs with different attribute values.

Attribute measurement (AM) represents the user's attribute information numerically; similarity measurement (SM) represents the similarity of keywords of the original blog posts among users and reflects the characteristics of users' spontaneous behaviors. User security degree (Sec),

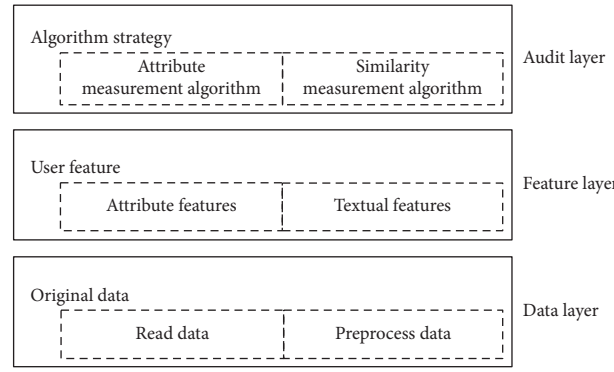


Figure 1: AM-SM-UAM framework.

which reflects the security degree of users, is calculated according to the user attribute measurement AM $(u)$ and published content similarity measurement SM $(u)$. The formula is shown as follows:

$$\mathrm{Sec}(u) = \mathrm{AM}(u) \cdot \mathrm{SM}(u). \tag{1}$$

*4.1. Attribute Measurement.* User attribute measurement is the basis for user security degree evaluation. The attributes of the measurement are shown in Table 1. In addition to the users' information integrity, other attribute information can be read in the experimental dataset, so the personal information integrity of users is defined and calculated.

Personal information integrity $(A_p)$ refers to the proportion of the personally valid information that the user has publicly filled out, which accounts for all the information to be filled out. All the information to be filled in includes 7 items such as microblog ID, real name authentication, gender, birthday, age, region, and company.

Personal information integrity was calculated, and the construction vector $E$ was used to represent the user's data, as shown in the following equation:

$$E = (d_1, d_2, d_3, d_4, d_5, d_6, d_7), \tag{2}$$

where $d_k$ $(k = 1, 2, \ldots, 7)$ indicates whether item $k$ is filled in completely, and $d_k = 0$ indicates that no valid information is filled in item $k$; $d_k = 1$ indicates that valid information has been filled in item $k$.

The user vector model was constructed. By obtaining the user's data, unmarked valid user tags were selected to judge the 7 data information, and the information was marked as valid or invalid according to the actual filling situation until all the user tags were marked. The user's information integrity is determined by calculating the scarcity of vector $E$, as shown in the following equation:

$$A_p(u) = \frac{1}{7} \sum_{k=1}^{7} d_k, \tag{3}$$

where $A_p$ $(u)$ represents the integrity of user $u'$s personal information; 7 is the total dimension of $E$.

According to the relative importance of the five user attribute information of microblog level $A_l$, big-V

Table 1: User attribute features.

| Feature symbol | Feature category | Feature name |
|---|---|---|
| $A_l$ | User attribute | Microblog level |
| $A_V$ | User attribute | Big-V certification |
| $A_p$ | User attribute | Personal information integrity |
| $A_f$ | User attribute | Number of followers |
| $A_s$ | User attribute | Number of fans |

certification $A_V$, personal information integrity $A_p$, number of followers $A_f$, and number of fans $A_s$, the hierarchical decision model was used to calculate the weight vector $\beta$, and the specific value is determined by experiments.

The structure of the hierarchical decision model includes the target layer, the criterion layer, and the scheme layer, as shown in Figure 2. The first layer represents the target layer of the metric user; the second layer represents the criterion layer that the five user attribute features affect the target determination, and the third layer represents the scheme layer of the user activity.

According to the attribute vectors corresponding to the five user features of microblog level $A_l$, big-V certification $A_V$, personal information integrity $A_p$, number of followers $A_f$, and number of fans $A_s$, and combined with the weight vector $\beta$, the user attributes are numerically represented to reflect the user's own security degree, as in the following equation:

$$\mathrm{AM}(u) = \left(A_l, A_v, A_p, A_f, A_s\right) \cdot (\beta)^T. \tag{4}$$

*4.2. Similarity Measurement.* Users' original blogs reflect their behavior features. Keywords in user blog content are extracted, and similarity of blog content among users with different attribute values is estimated to discover user behavior characteristics and complete user similarity measurement. The similarity of the blog can be converted into the problem of similarity between two strings, and the operation steps between strings are utilized for calculation.

Levenshtein distance refers to the minimum number of editing operations required to convert the source string into the target string between the source string and the target string [16], and the allowed to edit operation includes replacing, inserting, and deleting.

Since the user blog post appears in the form of long and short sentences, and the sequence of long and short sentences in a blog post does not influence the similarity of users, there are two disadvantages indirectly using the edit distance calculation. First, the experimental error of taking a whole blog post as a comparison string is large. Second, the number of substitutions of the sequence of long and short sentences in a blog post will be counted into the number of operations, increase the editing distance, and reduce the similarity, and has errors compared with the actual situation.

In this regard, two improvement methods of editing distance are proposed when constructing the similarity measurement algorithm. (a) Jieba [17] was used to process the user's blog content, dividing the whole post into several keywords. (b) The sequence of keywords in actual blog posts does not affect the judgment of similarity. To avoid the phenomenon of low similarity caused by inconsistent word order, the overlapping keywords in the two strings are deleted, and then the similarity measurement is carried out.

The similarity measurement algorithm steps are as follows:

Step 1: set up two sets of original keywords composed of keywords of blog contents, and name them, respectively, *keySetS* and *keySetT*, where the number of keywords is defined as the size of the set, named *keyNumS* and *keyNumT*.

Step 2: traverse the keywords in original keywords sets, get the coincidence keywords *keySame*, and delete them in sets, respectively. At the same time, record the number of coincident keywords named *SameNum*.

Step 3: record the current keyword sets *keySetX* and *keySetY* after deleting the coincident keywords, and convert the two sets into a source string *strX* and a target string *strY*. Set $x_1 \ldots x_m$ and $y_1 \ldots y_n$ representing them, respectively, where $m$ is the length of *strX* and $n$ is the length of *strY*.

Step 4: define $(m+1)\cdot(n+1)$ order D$[m][n]$, and save the minimum number of edit operations needed to convert *strX* to *strY*, as shown in equation (5).

Step 5: calculate the similarity SM of blog posts. The formulas are shown in equations (6) and (7).

$$D[m][n] = \begin{cases} 0, & m = 0, n = 0, \\ n, & m = 0, n = 0, \\ m, & m > 0, n = 0, \\ \min\{D[m-1][n] + 1, D[m][n-1] + 1, D[m-1][n-1] + flag\} & m > 0, n > 0, \end{cases} \tag{5}$$

where *flag* is used to mark the number of valid substitutions during the comparison of the *strX* and *strY* characters, $\text{flag} = \begin{cases} 0, X[m] = Y[n] \\ 1, X[m] \neq Y[n] \end{cases}$.

In equation (5), when $m > 0$ and $n > 0$, it corresponds to three operation modes of strings, respectively: (a) delete operation: D$[m-1][n]+1$ means to delete the last character of *strX* and add 1 to the number of editing; (b) insert

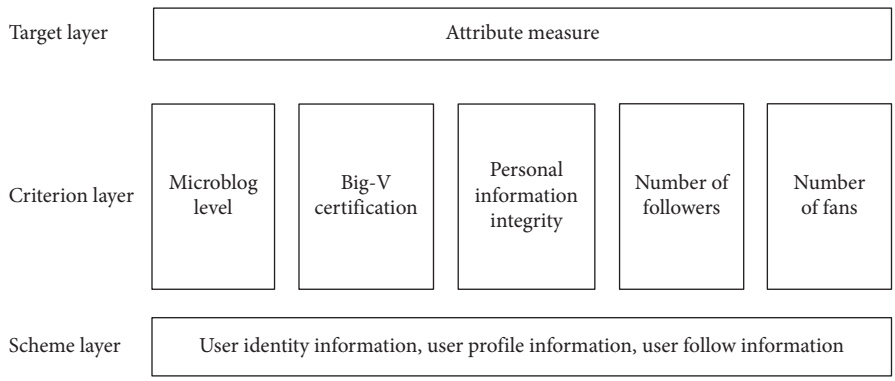


Figure 2: Hierarchical decision model.

operation: $D[m][n-1]+1$ means that the last character of *strY* is inserted into *strX*, and the number of editing is increased by one; (c) replace operation: $D[m-1][n-1]+flag$ indicates that the last character of the string *Y* is substituted to *strX*. The number of editing is determined by the *flag*, which is used to mark the number of valid substitutions:

$$\text{sim} = \left(1 - \frac{D[m][n]}{\max(m,n)}\right) + \frac{SameNum}{\max(keyNumS, keyNumT)}, \tag{6}$$

$$\text{SM} = \frac{\text{sim}}{\text{sim}_{\max}}, \tag{7}$$

where $D[m][n]$ represents the Levenshtein distance between the source string *strX* and the target string *strY*.

## 5. Experiments

*5.1. Experimental Environment and Data.* The environment used in the experiment was Intel(R) Core(TM) i5-7300HQ CPU @2.50 GHz, 8 GB of memory, the operating system is Windows 10, and Model code is based on C++ implementation.

The dataset published in [18] was used to verify the feasibility of the model. The dataset contains 1,787,443 microblog user data, and each user data includes basic information of the user (such as user ID, gender, number of followers, and number of fans) and 1000 microblogs newly released by each user. Among them, there are nearly 4 billion relationships of mutual concern among users. Due to a large amount of data in the dataset, 10 groups are randomly selected from the dataset, each group has 10,000 pieces of user data, and each piece of user data includes the basic information of the user and the newly published blog content, which is recorded as "Data1," "Data2," "Data3," "Data4," "Data5," "Data6," "Data7," "Data8," "Data9," and "Data10."

*5.2. Evaluation Index.* To solve the data imbalance problem, confusion matrix analysis experiment results were established [19]. In the matrix, TP stands for the number of users that are originally malicious users and are judged to be malicious users during detection; FN stands for the number of users that are originally malicious users but are judged to be normal users during detection; FP stands for the number of users that are originally normal users but are judged to be malicious users during detection; and TN stands for the number of users that are originally normal users and are judged to be normal users during detection, as shown in Table 2.

Table 2: Symbol description.

| Detection result | Actual situation | |
|---|---|---|
| | Malicious users | Normal users |
| Malicious users | TP | FP |
| Normal users | FN | TN |

To evaluate the performance of UAM, three evaluation indexes, namely, precision rate (Pre), recall rate (Rec), and harmonic mean value F1_score were selected. Among them, the precision rate and recall rate were used to evaluate the accuracy of the experiment, and the harmonic mean value was used to evaluate the comprehensive performance of the experiment, and the definitions are shown in the following equations:

$$\text{Pre} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{8}$$

$$\text{Rec} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{9}$$

$$\text{F1\_score} = 2 \cdot \frac{\text{Pre} \cdot \text{Rec}}{\text{Pre} + \text{Rec}}. \tag{10}$$

*5.3. Parameter Settings.* Parameters involved in the experiment include security threshold $\varphi$ and weight vector $\beta$. The safety threshold $\varphi$ was optimized through experiments, and its value was determined by referring to the performance of the model evaluated by means of harmonic mean F1_score, as shown in the following analysis; the weight vector $\beta$ is determined by a hierarchical decision model, and the calculation process is as follows.

According to the hierarchical model, user attributes are measured, in which $W_1$, $W_2$, $W_3$, $W_4$, and $W_5$ represents $A_l$,

Table 3: Judgment matrix.

| | $W_1$ | $W_2$ | $W_3$ | $W_4$ | $W_5$ |
|---|---|---|---|---|---|
| $W_1$ | 1 | 3/5 | 3/7 | 3 | 3 |
| $W_2$ | 5/3 | 1 | 5/7 | 5 | 5 |
| $W_3$ | 7/3 | 7/5 | 1 | 7 | 7 |
| $W_4$ | 1/3 | 1/5 | 1/7 | 1 | 1 |
| $W_5$ | 1/3 | 1/5 | 1/7 | 1 | 1 |

Table 4: Treated matrix.

| | $W_1$ | $W_2$ | $W_3$ | $W_4$ | $W_5$ | *Sum* | $\beta$ |
|---|---|---|---|---|---|---|---|
| $W_1$ | 1 | 3/5 | 3/7 | 3 | 3 | 8.029 | 0.163 |
| $W_2$ | 5/3 | 1 | 5/7 | 5 | 5 | 13.381 | 0.242 |
| $W_3$ | 7/3 | 7/5 | 1 | 7 | 7 | 18.733 | 0.463 |
| $W_4$ | 1/3 | 1/5 | 1/7 | 1 | 1 | 2.676 | 0.066 |
| $W_5$ | 1/3 | 1/5 | 1/7 | 1 | 1 | 2.676 | 0.066 |

$A_V$, $A_s$, $A_f$, and $A_h$ five attribute features of users. The weights of the five features are set as $W_1 = 3$, $W_2 = 5$, $W_3 = 7$, $W_4 = 1$, and $W_5 = 1$. The proportional nine scale method [20] proposed by T.L. Saaty is used as a comparison scale to compare the relative importance of each index in the criterion layer. The structural judgment matrix is shown in Table 3.

By calculating the weight vector $\beta$ of each attribute through the judgment matrix, *Sum* the matrix by row and normalize the vector Sum, as shown in Table 4.

The relative importance of the five attributes was obtained, and the weight vector $\beta$ was obtained as follows: $\beta =$ (0.163, 0.242, 0.463, 0.066, 0.066).

*5.4. Experimental Analysis.* To compare the performance difference between AM-SM-UAM and the existing advanced model, a comparative experiment was set up. AM-SM-UAM was compared with the DBSCAN-based clustering algorithm and PageRank-based anomaly detection algorithm. Through the three algorithms corresponding to the various indicators of the experiment, the accuracy of the three algorithms to identify malicious users of a microblog is analyzed.

The clustering algorithm based on DBSCAN is an anomaly detection method based on density clustering, which can find abnormal points while clustering. The PageRank-based microblog account anomaly detection algorithm constructs a social relationship matrix according to the user relationship and ranks the account by iteratively calculating the PageRank value to detect malicious users. Both algorithms have good results in malicious user identification, so the above two algorithms are used to compare experiments with AM-SM-UAM. Using these three algorithms, 10 groups of experiments were conducted on the dataset of "Data1-Data10" in turn, which were recorded as "G1-G10". Pre, Rec, and F1_score were used as the evaluation criteria of the experiment, and the experimental results are shown in Figure 3–5.

The results show that when AM-SM-UAM identifies malicious users, the precision rate difference between the 10 groups is no more than 2%, the recall rate is no more than 3%, and the F1_score is no more than 1%. Meanwhile, the precision rate, recall rate, and F1_score are all high. DBSCAN clustering algorithm and PageRank ranking algorithm have a lower precision rate when detecting malicious users of microblog, and the recall rate fluctuates

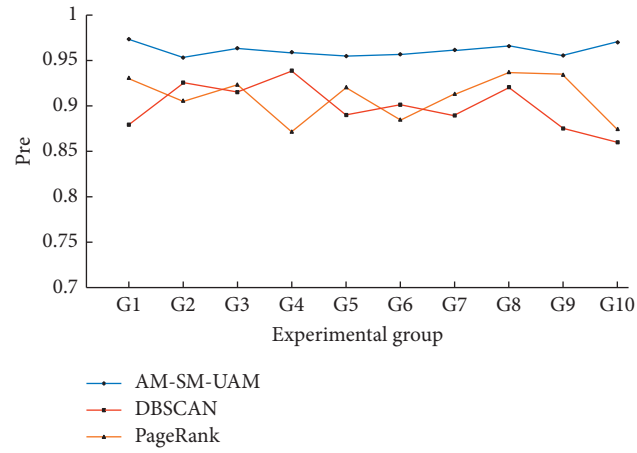


Figure 3: Precision rate.

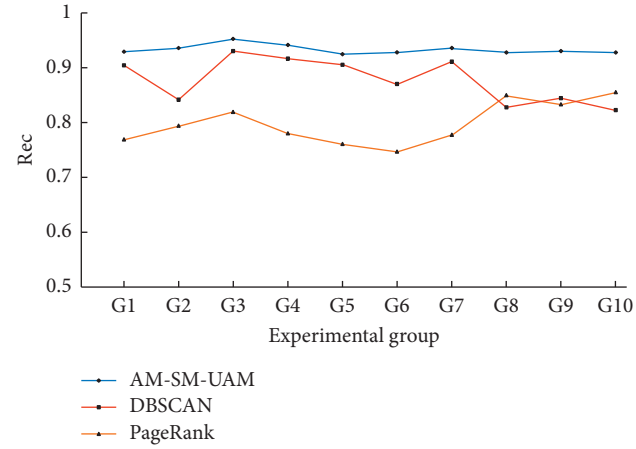


Figure 4: Recall rate.

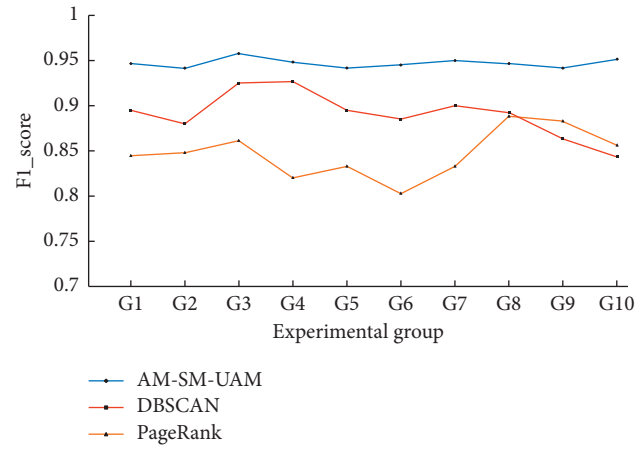


Figure 5: F1_score.

greatly, which makes the F1_score lower and unstable. According to the experimental results, the audit of users in microblog by AM-SM-UAM is completed based on the user's attribute information and the user's blog keywords. It not only considers the numerical information such as the user's attributes and reduces the influence of time delay caused by considering only the user's behavior, but also considers text information such as blog content, and the incompleteness caused by the calculation of only numeric attributes is avoided, thus improving the accuracy of identifying malicious users.

To test the stability of AM-SM-UAM audit microblog users, the average and variance of 10 groups of experimental results corresponding to the three algorithms were compared. The experimental results are shown in Figures 6 and 7.

It can be observed in Figure 6 that the 10 sets of experiments corresponding to the three algorithms are compared in terms of precision rate, recall rate, and F1_score value. Among them, the average value of the three indexes of the DBSCAN clustering algorithm is medium; the PageRank ranking algorithm although the average value is 92%, its recall rate is low, and the overall performance of the algorithm is poor. Among the 10 experiments using AM-SM-UAM, the precision rate, recall rate, and F1_score were the highest compared with the other two algorithms; the average accuracy can reach 96%.

As can be seen from Figure 7, the variance of the DBSCAN clustering algorithm and PageRank ranking algorithm on the three experimental evaluation indexes is large, indicating that the experimental results of the above two algorithms fluctuate greatly in the 10 groups of experiments, respectively, and the stability of the algorithm is poor. The variance of the 10 groups of experiments corresponding to AM-SM-UAM is small, indicating that the results of each group of experiments are less fluctuating and the stability of the algorithm is better.

According to the mean value and variance of the 10 groups of experimental results corresponding to the three algorithms, in the process of auditing microblog users' experiment, compared with the other two algorithms, AM-SM-UAM algorithm also has better stability and adaptability under the premise of ensuring a higher accuracy of identifying malicious users.

*5.5. Parameter Tuning.* DBSCAN clustering algorithm, PageRank ranking algorithm, and AM-SM-UAM algorithm all require parameter adjustment to achieve malicious user identification. The DBSCAN clustering algorithm needs to set two parameters, namely, neighborhood threshold (Eps) and point threshold (Minpts). According to the parameters, the region with a certain density is divided into clusters, and the clustering results are sensitive to the parameter values. The PageRank ranking algorithm calculates the user PR value by matrix iteratively to rank the user to complete the detection of the malicious user and the setting of the damping factor and the iteration termination threshold has a decisive influence on the user PR value calculation, and the ranking result is sensitive to the parameter value. The above two algorithms are greatly affected by the parameters, and the performance of the algorithm fluctuates greatly.

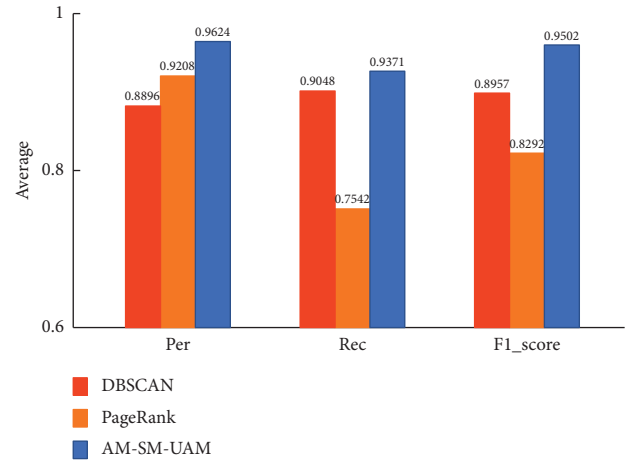


Figure 6: The average.

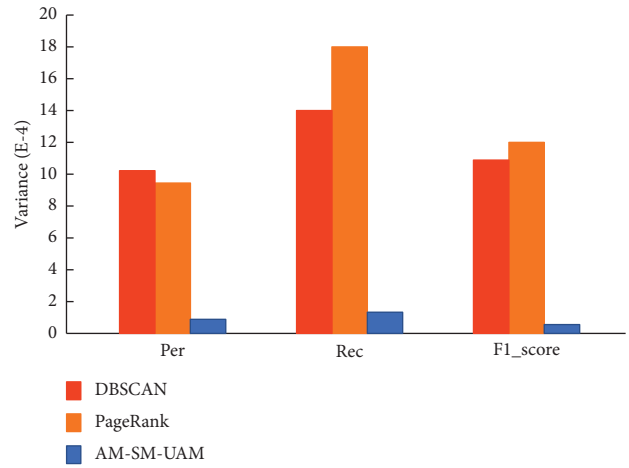


Figure 7: The variance.

The security threshold $\varphi$ in AM-SM-UAM is related to the accuracy of identifying malicious users. By equation (1), the formula of the safety of users for the $\mathrm{Sec}(u_i) = \mathrm{AM}(u_i) \cdot \mathrm{SM}(u_i)$, in which $\mathrm{AM}(u_i) = (A_l, A_V, A_p, A_f, A_s) \cdot (\beta)^T$ the weight vector of beta calculated by hierarchical decision model. Therefore, on the premise that the weight vector $\beta$ has been determined, the safety threshold $\varphi$ should be determined by the size of F1_score and the relationship between the security threshold $\varphi$ and F1_score is shown in Figure 8.

As can be observed in Figure 8, when the security threshold $\varphi$ is 0.4, the F1_score value is the largest. Therefore, when the security threshold $\varphi = 0.4$, that is, the user security degree less than 0.4 users defined as malicious users, AM-SM-UAM has the best performance.

To verify the rationality of the security threshold of 0.4, 10 groups of experiments of AM-SM-UAM auditing microblog users were analyzed. Take the user security degree of normal users and malicious users in microblog calculated from the "G1-G10" 10 groups of experiments, and respectively. calculate the average of the security degree of normal users and malicious users in each group of experiments, as shown in Figure 9.

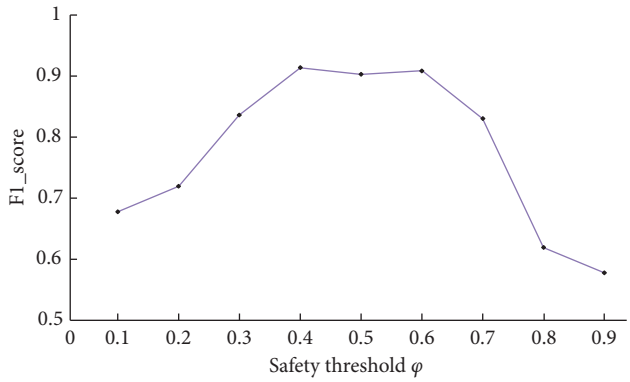


Figure 8: Relationship between $\varphi$ and F1_score.

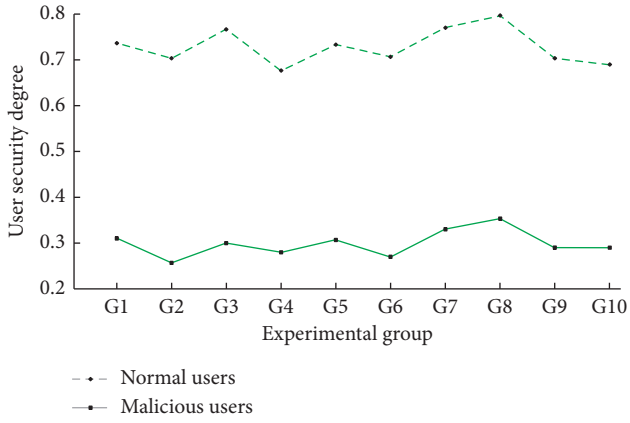


Figure 9: User security degree distribution.

The results show that the average security degree of normal users is distributed in [0.6, 0.8], while that of malicious users is distributed in [0.2, 0.4]. According to the experimental results, the average security degree between normal users and malicious users in microblog has a large gap, so the degree range of the security threshold can be [0.4, 0.6]. According to the experimental results, compared with the other two algorithms, in the process of parameter tuning, UAM is easy to find the optimal parameters, which is more conducive to identifying malicious users in the microblog.

## 6. Conclusion

This paper proposes a microblog user audit model based on attribute measurement and similarity measurement (AM-SM-UAM), which is used to detect a large number of malicious nodes in the IoT and identify false information on medical and social networks. Firstly, the concept of user security degree was proposed to reflect the security level of microblog users, as the standard of differentiation between malicious users and normal users. Secondly, the user attribute measurement algorithm was constructed, using a hierarchical decision model to construct a judgment matrix to analyze user attribute data. Finally, the similarity measurement algorithm was constructed, keywords of user original blog with word segmentation technology were extracted, Levenshtein distance was improved, user blog content similarity was calculated, and user behavior information data were analyzed. Through the measurement of the user attribute information and the calculation of the similarity of the blog keywords, the user security degree was obtained, and the malicious user um was determined. Experiments showed that AM-SM-UAM achieved more accurate and stable performance.

In the future, the behavior of malicious user nodes in the IoT will be specifically analyzed to determine the correlation behavior between malicious users. At the same time, the probability of associative behaviors between malicious nodes in medical IoT is considered by increasing inference calculation, and the identification of malicious nodes and false behaviors in medical IoT is further discussed.

## Data Availability

The data came from an article [18] by Zhang Jing of Tsinghua University, in which crawlers were used to construct a dataset of microblog users. The microblogging network they used in this study was crawled from Sina Weibo.com, which, similar to Twitter, allows users to follow each other. Particularly, when user A follows B, B's activities such as (tweet and retweet) will be visible to A. A can then choose to retweet a microblog that was tweeted (or retweeted) by B. User A is also called the follower of B and B is called the followee of A. After crawling the network structure, for each one in the 1,787,443 core users, the crawler collected her 1,000 most recent microblogs. At the end of the crawling, they produced in total 4 billion following relationships among them, with an average of 200 followers per user.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was supported by the National Key R&D Program of China under Grant no. 2017YFB0802300.

## References

[1] K. Fan, S. Sun, Z. Yan, Q. Pan, H. Li, and Y. Yang, "A blockchain-based clock synchronization scheme in IoT," *Future Generation Computer Systems*, vol. 101, pp. 524–533, 2019.

[2] K. Fan, W. Jiang, L. Qi, H. Li, and Y. Yang, "Cloud-based RFID mutual authentication scheme for efficient privacy preserving in IoV," *Journal of the Franklin Institute*, vol. 9, no. 35, 2019.

[3] M. S. Hajmohammadi, R. Ibrahim, A. Selamat, and H. Fujita, "Combination of active learning and self-training for cross-lingual sentiment classification with density analysis of unlabelled samples," *Information Sciences*, vol. 317, pp. 67–77, 2015.

[4] A. Gupta, P. Kumaraguru, C. Castillo, and P. Meier, "TweetCred: real-time credibility assessment of content on twitter," in *Proceedings of the 6th International Conference on*

*Social Informatics*, pp. 228–243, Springer, Barcelona, Spain, 2014.

[5] A. A. Amleshwaram, N. Reddy, S. Yadav, G. F. Gu, and C. Yang, "CATS: characterizing automation of Twitter spammers," in *Proceedings of the 5th International Conference on Communication Systems and Networks (COMSNETS)*, pp. 1–10, IEEE, Bangalore, India, 2013.

[6] X. Hu, J. L. Tang, and H. Liu, "Online social spammer detection," in *Proceedings of the 28th Conference on Artificial Intelligence*, AAAI, Quebec, Canada, pp. 59–65, July 2014.

[7] X. Hu, J. L. Tang, Y. C. Zhang, and H. Liu, "Social spammer detection in microblogging," in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pp. 2633–2639, AAAI, Beijing, China, 2013.

[8] S. Ravikumar, K. Talamadupula, R. Balakrishnan, and S. Kambhampati, "RAProp: ranking tweets by exploiting the tweet/user/web ecosystem and inter-tweet agreement," in *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*, pp. 2345–2350, ACM, San Francisco, CA, USA, 2013.

[9] K. Lee, J. Caverlee, and S. Webb, "Uncovering social spammers: social honeypots+machine learning," in *Proceeding of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 435–442, ACM, Geneva, Switzerland, 2010.

[10] J. Zhang, Y. Yang, H. Wang et al., "Semi-supervised clustering ensemble based on collaborative training," in *Proceedings of the International Conference on Rough Sets and Knowledge Technology*, Springer-Verlag, Berlin, Germany, pp. 450–455, 2012.

[11] N. Tahir, A. Hassan, M. Asif et al., "MCD: mutually connected community detection using clustering coefficient approach in social networks," in *in Proceeding of the 2nd International Conference on Communication, Computing and Digital Systems (C-CODE)*, IEEE, Islamabad, Pakistan, March 2019.

[12] Y. Meng and L. F. Kwok, "Intrusion detection using disagreement-based semi-supervised learning: detection enhancement and false alarm reduction," in *Proceedings of the International Conference on Cyberspace Safety and Security*, Springer-Verlag, Melbourne, Australia, pp. 483–497, 2012.

[13] J. Zhu, Y. Li, and R. Liu, "Social network group identification based on local attribute community detection," in *Proceedings of the IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, IEEE, Chengdu, China, March 2019.

[14] C. Ju, K. Zhao, and F. Bao, "Influence intensity calculation model of social network users integrating closeness centrality and credit," *Journal of Intelligence*, vol. 38, no. 2, pp. 170–177, 2019.

[15] S. Li, X. Li, H. Yang et al., "A zombie account detection method in microblog based on the pagerank," in *in Proceedings of the 2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, IEEE, Prague, Czech Republic, July 2017.

[16] X.-M. Lin and W. Wang, "Set and string similarity queries: a survey," *Chinese Journal of Computers*, vol. 34, no. 10, pp. 1853–1862, 2011.

[17] https://pypi.org/project/jieba/.

[18] J. Zhang, J. Tang, J. Li, Y. Liu, and C. Xing, "Who influenced you? predicting retweet via social influence locality," *ACM Transactions on Knowledge Discovery from Data*, vol. 9, no. 3, pp. 1–26, 2015.

[19] M. Yang, J. M. Yin, and G. L. Ji, "Classification methods on imbalanced data: a survey," *Journal of Nanjing Normal University (Engineering and Technology Edition)*, vol. 8, no. 4, pp. 7–12, 2008.

[20] Z. Q. Luo and S. L. Yang, "Comparative study on several scales in AHP," *Systems Engineering-Theory & Practice*, vol. 9, pp. 51–60, 2004.