Efficient Sensitive Data Identification Using Multi-Protocol Automatic Parsing and Vector Classification Based on Feature Fusion

Jian Hu[1*], Lina Zhang[1], Guangqian Lu[1], Wenqian Xu[2,3], Yixin Jiang[2,3]
[1]Yunnan Power Grid Information Center, China Southern Power Grid, Kunming 650217, China
[2]Electric Power Research Institute, China Southern Power Grid, Guangzhou 510663, China
[3]Guangdong Provincial Key Laboratory of Power System Network Security, Guangzhou 510663, China
E-mail: csghujian@126.com

[*]Corresponding author

Keywords: feature fusion, sensitive data, efficient identification, dataset

Received: September 27, 2024

At this stage, data has become essential and important information in people's daily life, and how to automate the identification and classification of sensitive attributes of structured datasets in the production environment, etc., has also become a key issue in data privacy protection. This paper proposes an efficient recognition technique for sensitive data based on feature fusion. The method can effectively improve the accuracy of sensitive data identification, realize the identification and classification of sensitive attributes, and take into account the correlation and association between attributes. Moreover, based on the artificial intelligence method, a high-precision and fine-grained identification technology for cross-domain encrypted traffic is constructed, which can support more than five abnormal data access behaviors that can be identified by the data security supervision system, and can support the recovery of traffic with business data not less than 500Mbps.

Povzetek: Predlagana metoda združuje večprotokolno samodejno analizo in vektorsko klasifikacijo z združevanjem značilk za učinkovito prepoznavanje občutljivih podatkov. Izboljšuje natančnost, podpira obsežne nize podatkov in zmanjšuje tveganje uhajanja informacij.

1 Introduction

Big data platform centralized storage of a large amount of data and user privacy information, once the data leakage, not only seriously jeopardize the user's privacy and personal information security, but also will cause huge economic losses and social impact on the telecommunications enterprises. How to establish a security protection system for sensitive data, make sensitive data manageable, controllable and traceable, ensure the normal operation of various businesses, and prevent data leakage is a serious challenge for operators. In this paper, we propose a feedback-based closed-loop DPI sensitive data detection method (Table 1), which can be effectively applied to the process of efficient sensitive data identification and auditing in the big data environment.

Table 1: Algorithm comparison

Method	Data set	Evaluation indicators	Key performance	Quote
Dictionary learning method	TF-IDF	Determine attributes based on their frequency of occurrence	Form a weighted sensitive vocabulary library and intelligently match sensitive data	[1]
Decision Tree Algorithm	ID3	Not dependent on sensitive data dictionary	Record the frequency, region, and sensitivity level of sensitive words, and finally give the weight of sensitive words to identify sensitive data	[2]
Probability theory algorithm	K-Means	Probability of occurrence in sensitive and non sensitive datasets	Privacy breaches may occur when calculating the distance between data points and cluster center points	[3]
Feature Fusion Recognition Method	DPI	By establishing a session table, real-time recording of the identified status is achieved	Search and match pre-defined keywords or feature from the payload content of data packets, and process the data packets based on the matching results	this paper

2 Technical implementation ideas

2.1 Automated protocol feature extraction and parsing

2.1.1 Multi-protocol automated extraction model

Adopting Apriori algorithm is used for frequent itemset mining and association rule learning on relational databases, utilizing an iterative method of layer-by-layer search traversal to find out the relationship of itemsets in the data and form the rules, the process is composed of joining (class matrix operation) and pruning (removing the unnecessary intermediate results). The itemset contains K itemsets and the frequency of occurrence of the itemset is the number of transactions containing the itemset, called the itemset frequency. If an itemset satisfies minimum support, it is a frequent itemset. Frequent itemsets determined by Apriori can be used to identify association rules that highlight general trends in the database: areas such as market basket analysis. Frequent itemsets are mined by candidate set generation downward closure detection. The iterative layer-by-layer search approach has a straightforward algorithm with no complex derivations and is easy to implement.

2.1.2 Multi-protocol automated parsing model

Apriori algorithm is used to frequently mine itemsets and learn association rules in relational databases. The relationship between itemsets in data is found by iterative method of searching and traversing layer by layer, and rules are formed. The process consists of joining (matrix-like operation) and pruning (removing unnecessary intermediate results). There are k itemsets in itemsets, and the occurrence frequency of itemsets is the number of transactions containing itemsets, which is called itemset frequency. If an itemset meets the minimum support, it is a frequent itemset. The frequent itemsets determined by Apriori can be used to determine the association rules that highlight the overall trend of the database: such as market basket analysis and other fields. Mining frequent itemsets through candidate set generation and downward closure detection. The iterative method of layer-by-layer search has the advantages of simple algorithm, no complicated derivation and easy implementation. Compared with FP-Growth algorithm, it has good accuracy and adaptability. FP-Growth can effectively compress the transaction database, without repeated traversal and scanning, and without generating candidate sets, although it effectively solves the efficiency defects in Apriori algorithm. However, the frequent pattern tree FP-Tree constructed by this algorithm needs to occupy a large amount of memory. When the transaction database is relatively large, it is difficult to implement this method. Therefore, the multi-protocol automatic extraction model can effectively solve the above problems [4].

2.1.3 FDN method

The overall model structure of FDN is similar to the CGC structure, including several parts of bottom, feature combination and task, in which the bottom part performs feature decomposition, and a single or multiple group decomposition pair (DeComposition Pair, abbreviated as DCP) is used for each task, and each group of decomposition pairs includes two types of feature characterizations, task-sharing and task-unique, and in the feature combination part combines the task-sharing and task-unique parts to get the input of the task for the learning of the subsequent task parts. The learning of the model for each task can be expressed as equation (1), where the input of the task consists of the combination of two feature representations, shared and exclusive, $f^{\delta}(x)$ denoting the feature representation shared by the task in a certain task DCP, $f^{p}(x)$ denoting the feature representation exclusive to the task, and $g_k(\cdot)$ denoting the way of combining the feature representations by the task K, σ as the activation function.

$$f_k^{MTL}(X) = \sigma\left(g_k\left(f^{p}(X), f^{\delta}(X)\right)\right) \tag{1}$$

2.1.4 Feature representation orthogonality

The purpose of the orthogonality constraint is to make the two feature representations, shared and unique, in the DCP of each task as differentiated as possible, so as to improve the model's ability to capture the commonalities and differences between individual tasks.

$$L_{orth} = \sum_{k=1}^{K}\sum_{m=1}^{M}\left\|\left(f_m^{\delta}\right)^T f_m^{p}\right\|_F^2 \tag{2}$$

The computation of the orthogonality constraint is shown in equation (2), where K denotes the number of tasks, M denotes the number of feature decomposition pairs (DCPs) of task k, f_m^{δ} denotes the feature representation shared by the mth task of task k, and f_m^{p} denotes the feature representation unique to the mth task of task k, and $\|\cdot\|_F^2$ is the Frobenius paradigm, which is specific to matrices, and can be analogized to the vectorial L2 paradigm, which is computed by summing the squares and then squaring each element in the matrix [5].

2.1.5 Auxiliary task

Orthogonality constraints can strengthen the orthogonality of the two feature representations, while the orthogonality of the dimensions can be very much, cannot guarantee that the two are orthogonal to the dimension of task shared and task unique, so the introduction of auxiliary task constraints, so that the two representations of the DCP of the task, not only in the theoretical meaning of "shared" and " independent" in the theoretical sense, but actually learned on these two subspaces as well.

For each task, the independent feature representation of the task in DCP is regarded as a separate small network for the task feature extraction, so this part of the feature representation is used for task prediction as an auxiliary task. Taking the loss of auxiliary task as the constraint of task's unique feature representation learning can make this part of the representation focus on learning the characteristics of task, and combining the orthogonal constraint and the main loss constraint, the task sharing feature representation can focus on capturing the commonness of task.

$$L_{aux}=\sum_{k=1}^{K}\sum_{m=1}^{M}L_{k,m}(\hat{y}_m^k, y^k) \qquad (3)$$

The auxiliary task constraints are shown in equation (3), where $L_{k,m}(\cdot)$ is the auxiliary loss function of task k, which can be chosen to be consistent with the main loss, y^k is the true label of task k, and $\hat{y}_m^k$ is the task prediction of the mth unique feature representation in task k. The computation is shown in equation (4).

$$\hat{y}_m^k=\sigma\left(f_m^{k,p}(X)\right) \qquad (4)$$

2.1.6 Task shared representation fusion

The shared representations of all tasks are fused to obtain an overall shared representation, and each task concatenates it with the task's respective unique representations to obtain the task's input. The fusion operation of shared representations in this process can be seen as a constraint on the shared representations of each task, i.e., constraining the learning of shared representations of each task to the same subspace.

The overall loss of FDN consists of three parts: the main loss of the task, the orthogonal loss of the feature representations and the auxiliary task loss, as shown in equation (5).

$$L=L_{task}+L_{orth}+L_{aux} \qquad (5)$$

The core improvement of FDN is that it strengthens the constraints on feature representations so that it can extract features more in line with the original design intention [6]. When there is only the mainloss constraint, the feature representations are formally decomposed, but the actual learned may be cross and the differentiation needs to be improved. In this paper, the feature representations of the three methods of CGC, PLE and FDN are visualized as shown in Fig.1. It can be seen that by adding more constraints, the differentiation and differentiation of the two types of feature representations, task-sharing and task-exclusive, are significantly improved.

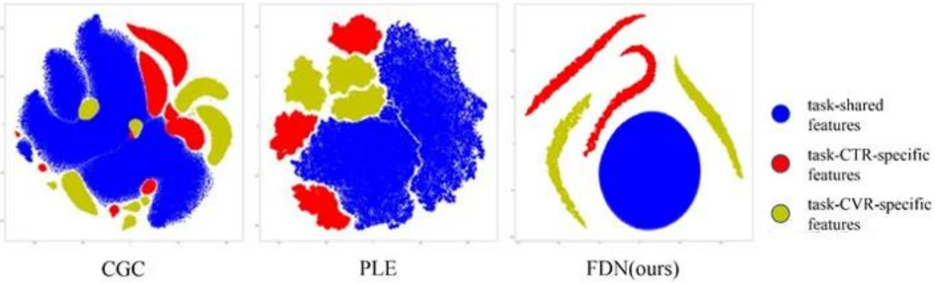

Fig. 1 Characteristics of the three methods of CGC, PLE and FDN

2.2 Fast screening of sensitive data traffic

Fast screening of sensitive data traffic is a context-based regular sensitive information detection method. The extracted sensitive information is screened by adding a contextual feature library. The feature library is divided into four types: front-added feature library, back-added feature library, front-subtracted feature library, and back-subtracted feature library.

As the name suggests, for example, front plus feature library, if the extracted information, such as identity card number, the initial score is set to 50, then, in the position extracted to the specified number of bytes before the appearance of the front features, the initial score will be added to the corresponding set of front score. For example: ID card number: 110xxxxxxxxxxx, you can "ID card number" in the first 10 bytes, plus the value of the score, as a pre-scoring features, and finally the initial score (50), plus the pre-scoring (eg: 40), 50 + 40 = 90, if 80 as a threshold, then the extraction to the of the ID number will be very credible [7].

Four feature libraries, used at the same time, compared the calculated score with the threshold, greatly improving the accuracy of the extracted sensitive information.

2.3 Sensitive data testing

2.3.1 Test environment hardware and software configuration

Table 2: Test environment hardware and software configuration

Host Usage	Quantity	Hardware Resources	System Software	IP Address
Fabric Mixer	1	CPU: Intel(R) Xeon(R) CPU E5-2698 v3 @ 2.30GHz 2*16C*2T internal storage:256G Intranet bandwidth：1000Mbps	CentOS 7	172.16.0.233

2.3.2 Testing tools

Table 3 Testing tools

Serial number	Name of tool	Description of use	Version
1	Jmeter	Analog Stressors	5.2.1
2	Prometheus+Granfana	Monitoring Resources Visualization Platform	2.22.2
3	Node_exporter	Monitor Server Resources CPU/Memory/Network/Disk	1.0.1
4	Process_exporter	Monitor process resources	0.4.0
5	Kafka_Exporter	Monitor queue access/consumption frequency and queue backlogs	1.2.0
6	InfluxDB	Performance data collection timing library	1.8.3
7	Jprofile+Jconsole	Monitor JVM memory and analyze	11.1.4
8	JMX_exporter	JMV probe data collection	0.1.2
9	Clickhouse_exporter	Monitor CH metrics collection	0.1.0
10	Redis_exporter	Monitor redis metrics collection	1.37.0

Using Jmeter at close to 50000tps pressure on httptopic initiated pre-buried about 4000-5000W or so of data, followed by turning on trsApllication to run out of spending power inflection point [8].

2.3.3 Test results

Table 4: Test results

Round	No sensitization	Mobile phone number sensitization	Bank card sensitization	Flow labeling [with sensitization]	Risk strategy [triggered]
First round	98%	0.8%	0.8%	0.2%	0.2%
Second round	95%	2%	2%	0.5%	0.5%
Third round	90%	4%	4%	1%	1%

During the test process, with the rise of sensitive data, 2% [the first round] 5% [the second round] consumption capacity does not change significantly. Only CPU consumption rises with the rise of sensitive data, 10% of the consumption capacity began to decline, because the current environment bandwidth has been dipped full of network bottlenecks, by the environment is limited by the program cannot continue to measure its own upper limit.

Table 5: Stress testing of sensitive data

Round	trs-efm-app (topic: httpTopic)	trs-output-pet (topic: k1_record_app)	K1-risk	CPU utilization
First round	49800tps	994tps	200tps	61%
Second round	50400tps	2520tps	505tps	67%
Third round	45200tps	4500tps	901tps	70%

The pressure test process involving sensitive data accounted for 5%, the overall data access and consumption trend is smooth trsapplication processing capacity basically remained at 11000tps, trsoutput processing capacity basically remained at 550tps or so, into the library to meet the expected percentage [9].

Pressure lasts 4 days in the process, the server each resource is smooth, the CPU is basically maintained at about 50%, memory 12.10 22:00 there is a slight increase (jmeter sensitive accounted for abnormalities caused by the discovery of restarting the jmeter after restoration of normal) [the overall basic maintenance of the stabilization of 124G], based on the overall trend of the overall trend is basically smooth, the fluctuation of the ignored controllable, the process is basically the same to observe. JVM trend is smooth without GC anomaly.

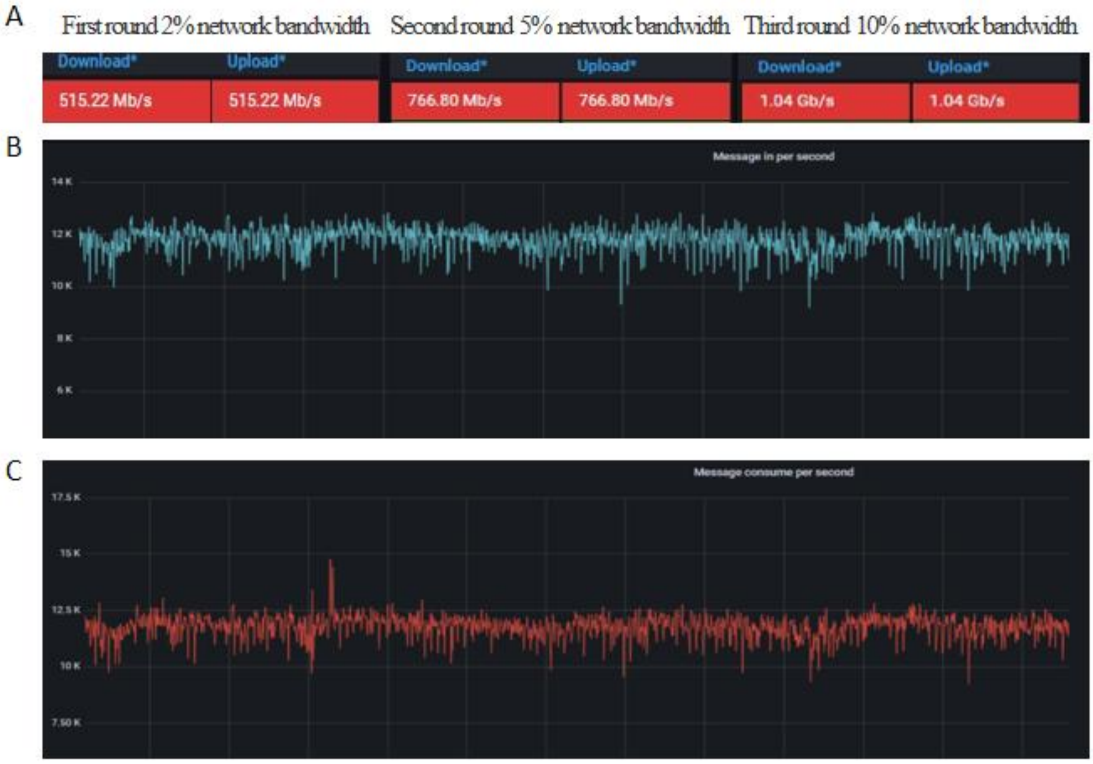

Figure 2: Shows the fluctuation situation. A: Proportion of sensitive data. B: Stable trend. C: Overall trend

As shown in this test, in the 32C64T 256G server gigabit network test environment, after many rounds of optimization, the current base flow engine 2.0 in respectively 2%, 5%, 10% of data pressure involving sensitive data ratio data, processing an average of 1.3KB data messages, the overall consumption capacity and into the library capacity to meet the expectations of not less than 20,000TPS indicators (Fig.2). And the overall server resource utilization also meets the industry's expected standards, and the CPU and memory resource trend is smooth during the stability process, with no memory leakage.

3 Efficient recognitions of sensitive data based on feature fusion

3.1 Multi-dimensional indicator judgment engine

Provides a variety of dimensional judgment indicators:

including 1: data content dimension; 2: metadata dimension; 3: statistical indicators dimension; 4: data feature fingerprint dimension.

Multi-dimensional indicator judgment engine provides custom complex indicator relationship expression way, used to meet the business data features need to be judged by multiple indicators based on complex combination of requirements. The rule expression form is as follows: (1 && 2) || (3 && (4 || 5)), where 1-5 are the five indicators defined in the multidimensional indicator engine.

Multi-dimensional indicator engine introduces data feature categories; the current multi-dimensional indicator engine defines three major categories: business features -> basic features -> original features; different categories of data features in the identification of different priorities in the hit in the multi-dimensional indicator engine decision-making logic to use [10-12].

3.2 Vectorized classification inference engine

3.2.1 Classification inference engine execution flow

Vector classification inference engine: to solve the problems of data table classification inference and field data type determination.

1) The engine receives a set of data features from the multi-dimensional index judgment engine.

2) The engine constructs a vector space based on the defined classification criteria and the data features associated with the data types in the criteria.

3) Based on the input data feature information, calculate the distance between the input data feature combination and each classification based on Euclidean distance, select the classification with the closest distance, and forward infer the classification to which the table belongs.

4) When multiple data types are possible for a certain data, based on the inferred classification, the data type of the data is inferred in the reverse direction.

3.2.2 Data classification methods

The vectorized classification engine proposes a method to analyze the data classification of data tables quantitatively by vectorizing the classification criteria and constructing a vector space [13].

The n-dimensional Euclidean space: the set consisting of the whole of an ordered array of n (positive integers) real numbers is called the n-dimensional point set or the n-dimensional Euclidean space, denoted as Rn. The construction of the vector space of the classification criteria is based on this theory, and the classification criteria are generalized into a vector space.

3.2.3 Calculate the classification based on recognized data features

1) The classification distance is calculated using the Euclidean distance, defined as follows: Any two points in Rn,$x = (x_1, x_2, \dots, x_n)$， $y = (y_1, y_2, \dots, y_n)$, then the Euclidean distance between these two points is defined as:$\rho(\chi, y) = \left[\sum_{i=1}^{n}(x_i - y_i)^2\right]^{\frac{1}{2}}$.

2) The input data features are permutated and all possible permutations are output, for each permutation the distance is computed with each classification respectively

3) Compare the results of each calculated distance and finally select the classification with the smallest distance and its corresponding data features [14].

3.3 User decision automatic feedback engine

3.3.1 User decision engine process

User decision automatic feedback engine: automatically identify the data characteristic parameters that need to be calibrated through the user's verification action, thus improving the accuracy of subsequent identification.

a) After the scanning of sensitive discovery tasks is completed, the system can carry out verification, and the user can modify the data types identified by the scanning again.

b) According to the user verification operation, find the original data type corresponding to the hit data feature a, the user selected data type corresponding to the hit data feature b.

c) Adjust the confidence (identifying the degree of confidence in the definition of data features) values of data features a and b to meet the similarity of data feature b is greater than the similarity of data feature a.

d) Based on the confidence level of the modified data feature, recalculate the data feature similarity of the field that has these two data features in the list of unverified and hit data features, determine the data feature of the field based on the calculation results, and perform classification calculations [15].

3.3.2 Data feature confidence adjustment method

□·feature similarity = data feature hit rate * data feature confidence level

□· has two data features a and b, the original data feature a similarity is greater than b, adjust the confidence level so that the data feature b similarity is greater than the data feature a similarity.

□·Adjustment of data features is prioritized to reduce the confidence of the data features with high similarity, the confidence can be reduced to a minimum of 1%, if the similarity has been reduced to 1% and still cannot meet the conditions, then increase the original similarity of the data features of the implementation of the low degree of confidence can be increased to a maximum of 100%. Take a with b features as an example: a similarity: Sa, hit rate: Ha, confidence level for Za, b similarity: Sb, hit rate: Hb, confidence level for Zb.

1) Decrease the confidence level Za such that Ha * Za < Sb, yielding Za< Sb/Ha.
2) If the calculation in the first step yields Za > 1, set Za to the value calculated in the first step; if Za is less than 1, set Za to 1 and proceed to the third step.
3) Elevate the confidence level Zb so that Ha * 1 < Hb * Zb, which results in Zb > Ha/Hb, if Zb is less than 100, set Zb to Ha/Hb, otherwise set Zb=100.

Enhance the accuracy of sensitive data identification, data classification should not infer classification from data features, but should mine the data feature set from classification [16]. Through the discovery of the three engines, the multi-dimensional indicator determination engine to identify data features, through the vectorized classification to infer the type determination, and then through the automatic feedback mechanism of user decision-making, to improve the accuracy of discovery and identification [17].

4 Conclusion

A large amount of sensitive information is stored in the data set for data release, if this sensitive information is not accurately identified and protected, once the information is leaked, it will not only cause losses to individuals, but also have a serious impact on the reputation of the enterprise. In the research, the rules of sensitivity clustering and association between attributes are mined, the attribute sensitivity defined by information entropy is established, and the automatic identification and classification algorithm of sensitive attributes is proposed to improve the classification and classification of sensitive attributes. It is found that the proposed algorithm can identify, classify and grade sensitive attributes of arbitrary structured data sets. Based on the above research on high-speed resolution processing technology for sensitive data traffic, it establishes high-precision and fine-grained abnormal access behavior identification technology for cross-domain encrypted traffic based on artificial intelligence methods, which can support the data security monitoring system to identify no less than five abnormal data access behaviors and support business data restoration traffic of no less than 500Mbps.

Funding statement

This work was sponsored in part by China Southern Power Grid Corporation technology project (059300KC2212000)

References

[1] Mendes R, Vilela JP. Privacy-Preserving Data Mining: Methods, Metrics and Applications. IEEE Access, 2017, 5: 10562-10582.
doi: 10.1109/ACCESS.2017.2706947

[2] Yoon T, Park SY, Cho HG. A smart filtering system for newly coined profanities by using approximate string alignment. Proc of IEEE International Conference on Computer & Information Technology. 643-650.
doi: 10.1109/CIT.2010.129.

[3] Srinivas B, Ramesh G, Sriramoju SB. An Overview of Classification Rule and Association Rule Mining. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 2018, (3): 1692-1697.

[4] Wang Z, Yang N, Chen Q, et al. A Blind Batch Encryption and Public Ledger-Based Protocol for Sharing Sensitive Data. China Communications, 2024, 21(1): 310-322.
doi: 10.23919/JCC.fa.2020-0640.202401

[5] Shang K, He W, Zhang S. Review on Security Defense Technology Research in Edge Computing Environment. Chinese Journal of Electronics, 2024, 33(1): 1-18.
doi: 10.23919/cje.2022.00.170

[6] Chen Q, Ye A, Zhang Q, et al. A New Edge Perturbation Mechanism for Privacy-Preserving Data Collection in IOT. Chinese Journal of Electronics, 2023, 32(3): 603-612.
doi: 10.23919/cje.2021.00.411

[7] Guan Z, Yang W, Zhu L, et al. Achieving adaptively secure data access control with privacy protection for lightweight IoT devices. Science China (Information Sciences), 2021, 64(6): 144-157.
doi: 10.1007/s11432-020-2957-5

[8] Tang J, Li R, Wang K, et al. A Novel Hybrid Method to Analyze Security Vulnerabilities in Android Applications. Tsinghua Science and Technology, 2020, 25(5): 589-603.
doi: 10.26599/TST.2019.9010067

[9] Liang S, Zhang Y, Li B, et al. SecureWeb: Protecting Sensitive Information Through the Web Browser Extension with a Security Token. Tsinghua Science and Technology, 2018, 23(5): 526-538.
doi: 10.26599/TST.2018.9010015

[10] Yan X. A Face Recognition Method for Sports Video Based on Feature Fusion and Residual Recurrent Neural Network. Informatica, 2024, 48(12): 137-152.
doi: 10.31449/inf.v48i12.5968

[11] Dong X, Li R, He H, et al. Secure Sensitive Data Sharing on a Big Data Platform. Tsinghua Science and Technology, 2015, 20(1): 72-80.
doi: 10.1109/TST.2015.7040516

[12] Rastogi S, Choudhary S. Face Recognition by Using Neural Network. Acta Informatica Malaysia. 2019; 3(2):07-09.
doi: 10.26480/aim.02.2019.07.09

[13] Qin X, Zhu X, Yang F, et al. Analysis of sensitive spectral bands of Tiangong-1 hyperspectral data for detecting fire status. Spectroscopy and Spectral Analysis, 2013, 33(7): 1908-1911. https://www.gpxygpfx.com/EN/10.3964/j.issn.1000-0593(2013)07-1908-04

[14] Han J, Yu J, Yu H, et al. Personalized privacy protection for sensitive values. Journal of Electronics, 2010, 38(7): 1723-1728.
doi: 10.62517/jike.202304417

[15] Wang L, Bo L, Jiao L. Density-sensitive spectral clustering. Journal of Electronics, 2007, (8): 1577-1581.
doi: 10.3321/j.issn:0372-2112.2007.08.030

[16] Shao C, Huang H, Zhao L. P-ISOMAP: A new data visualization algorithm that is less sensitive to neighborhood size. Journal of Electronics, 2006, (8): 1497-1501.
doi: 10.3321/j.issn:0372-2112.2006.08.028

[17] Guo Y. Research on sensitive data identification method. Information Recording Materials, 2017, 18(9): 89-91.
doi: 10.16009/j.cnki.cn13-1295/tq.2017.09.055