Privacy Protection Workflow Publishing Under Differential Privacy

Ning Wu[1,2], Jiaqiang Liu[1,2], Yunfeng Zou[1,2(✉)], Chao Xu[1,2], and Weiwei Ni[1,2]

[1] State Grid Jiangsu Electronic Power Company Research Institute, Nanjing 210036, China
yunfeng.zou@163.com
[2] Department of Computer Science and Engineering, Southeast University, Nanjing 211189, China

Abstract. The workflow has been widely used in data quality assessment, error data location and other fields. As data sharing deepens, so does the need to share data lineages. The topology of the lineage workflow contains private information that includes the data generation process, that is, the privacy of the lineage workflow structure, which directly exposes the structure privacy leakage of the lineage workflow. There are the following deficiencies in the privacy protection methods of the lineage workflow structure: (1) The privacy protection method based on the restricted release has a weak theoretical foundation and can only qualitatively measure the privacy protection effect of the lineage workflow structure; (2) Focusing on the maintenance of the local mapping relationship of modules, the maintenance of the key path of the lineage workflow is weak. Aiming at the above problems, this paper proposes a privacy protection method PPWP-DP for the lineage workflow structure, which satisfies the differential privacy. Key path and key path priority concepts are introduced. On this basis, the θ-project projection algorithm is proposed to reduce the degree of the lineage workflow. At the same time, according to the user's preference for key path priority, the maintenance of high priority key path reachability is achieved. The concept of oi-sequence is introduced to extract the structure characteristics of the lineage workflow, and add Laplacian noise to the oi-sequence to satisfy the differential privacy constraint. Adjust the global sensitivity of the oi-sequence after noise addition by the θ-project algorithm to reduce the Laplacian noise scale. Finally, the perturbed oi-sequence is used to reconstruct the lineage workflow for publication, which realizes the workflow privacy security and the maintenance of key path accessibility. Theoretical analysis and experiments verify the effectiveness of the proposed algorithm.

Keywords: Lineage workflow · Privacy security · Structural privacy · Differential privacy

W. Ni et al. (Eds.): WISA 2019, LNCS 11817, pp. 382–394, 2019.
https://doi.org/10.1007/978-3-030-30952-7_38

1 Introduction

Data lineage describes the generation principle and evolution process of data [1]. The data lineage plays an important role in data quality assessment, scientific experimental data reproduction, error data location, and data recovery [2, 3]. With the increasing widespread distribution of lineage workflows, the issue of privacy protection for lineage workflows is increasingly attracting researchers' attention, and the owners of lineage workflows want to protect certain sensitive information from potential attackers. Sensitive information in the lineage workflow can be divided into three categories [3–5]: (1) Intermediate data and parameter information during the execution of the lineage workflow; (2) Input-output mapping relationship of modules in the lineage workflow; (3) Topological relationship among modules in the lineage workflow. The above three types of sensitive information correspond to data privacy, module privacy and structural privacy of the lineage workflow [6].

At present, the privacy of the lineage workflow structure mainly adopts the protection method based on the deletion edge and aggregation module [7]. The main idea is to delete sensitive dependencies or aggregation modules in the lineage workflow according to the background knowledge of the workflow owner to achieve the purpose of protecting the privacy of the structure, and in the process of deleting sensitive dependencies, focusing on maintaining the local mapping relationship of the module input and output; The following defects existing:

(1) Based on the restricted release method of hiding sensitive dependencies, the theoretical basis of privacy processing technology is weak, and there is no strict privacy model and mathematical theory support. Causing the inability to quantitatively measure the privacy and security of the lineage workflow structure;
(2) The existing methods mainly maintain the local mapping relationship between the input and output of the module, and lack the maintenance of the key path describing the evolution process of the data item, which result in the poor availability of workflow.

To solve above problems, a lineage workflow publishing method PPWP-DP based on differential privacy is proposed. Based on the concepts of key path and key path priority, the θ-project algorithm is designed to reduce the degree of the lineage workflow, and the high-priority key path reachability is maintained according to the user's preference for key path priorities. The concept of oi-sequence is proposed and extract the structure characteristics of the lineage workflow. The global sensitivity of the oi-sequence after adding noise is adjusted by θ-project, so that the oi-sequence after the disturbance satisfies the ε-differential privacy model constraint while reducing the noise scale. Finally, the perturbed oi-sequence is used to reconstruct the lineage workflow for publication, which realizes the protection of workflow structure privacy and the maintenance of key path accessibility.

The main contributions of the paper are as follows:

(1) Introduce the concept of key path and key path priority. On this basis, the θ-project projection algorithm is proposed to reduce the degree of the lineage workflow. At the same time, according to the user's preference for the key path priority, the high

priority key path reachability is maintained. The concept of oi-sequence is devised to realize structure feature extraction from lineage workflow.

(2) The θ-project algorithm makes the degree of the module in the lineage workflow not greater than θ, reduces the global sensitivity of the oi-sequence, as well as scale of Laplacian noise added to the oi-sequence while satisfying the ε-differential privacy constraint.

(3) This paper proposes a privacy protection method PPWP-DP for the lineage workflow structure that satisfies the differential privacy model constraint, and realizes the purpose of quantitatively measuring the privacy security degree of the lineage workflow through ε value; And realizing the release of the oi-sequence reconstructed lineage workflow after the disturbance, while maximizing the accessibility of the key path; verifying the effectiveness of the proposed method through experiments on the public dataset.

The paper is organized as following: The second chapter introduces the research status of the privacy protection of the lineage workflow in recent years. The third chapter describes the problem and proposes the solution. The fourth chapter introduces the concept of the key path and oi-sequence of the lineage workflow, and proposes the privacy release method of the lineage workflow structure based on the ε-differential privacy. The fifth chapter analyzes and verifies the effectiveness of the proposed method. Finally, summarizing the full text and looking forward to the follow-up work.

2 Related Work

In recent years, the protection of lineage workflow privacy has received continuous attention, and researchers have done a lot of work in the field of lineage workflow privacy protection. Literature [3] proposes a module privacy protection model based on L-Diversity, which generates a visible view by hiding part of the attribute set, so that for any set of inputs, the attacker can guess that the probability of correct output of the module is not more than 1/Γ. Literature [7] uses the technique of deleting sensitive edges and aggregating adjacent modules to protect the sensitive edges of the lineage workflow. Literature [8] uses differential privacy protection technology to implement module privacy protection by adding noise attributes. Literature [9] proposes a method of privacy protection for the lineage workflow based on view and access control. First, dividing the roles of workflow users and generating different security views for each role to achieve privacy protection. [10] proposed a semantic-based workflow protection mechanism for the lineage, which realizes the representation and execution of the privacy mechanism by defining privacy protection and related analysis terms as ontology. [11] proposed a hierarchy protection workflow protection framework based on semantic representation and distributed execution. [12] introduced the concept of closure in the lineage workflow, and solved the problem of private information dissemination in the lineage workflow containing both public and private modules by hiding the closure between private groups.

The differential privacy model has been widely used in the field of graph privacy protection with good privacy security. The application of the differential privacy model

in the graph is mainly divided into two aspects [19]: edge differential privacy and node differential privacy. Since the global sensitivity of the node difference is too large, the edge difference is wider than the node difference in the field of graph privacy protection. The dk-graph model [17] can effectively extract the structural features of the graph, and has a central role in the edge differential privacy model protection; but when d is greater than 3, the global sensitivity is large, resulting in poor usability of the graph. [20] proposed a differential privacy model protection technique based on sampling and smoothing sensitivity, which reduces the global sensitivity of the differential privacy model by sampling and reduces the noise scale.

3 Problem Description and Related Concepts

3.1 Problem Description

The existing privacy protection methods for the lineage workflow structure are mainly based on the privacy protection technology of the deletion edge and aggregation module. The privacy protection technology based on the deletion edge and aggregation module is simple, lacks strict privacy model and mathematical principle support, and the theoretical foundation is weak. It is difficult to quantitatively measure the security level of the lineage workflow after privacy protection. The evolution of certain data items in the lineage workflow is critical to the user and is referred to as the key path. Maintaining the accessibility of key paths in the privacy protection of the lineage workflow structure is important for improving the availability of the lineage workflow. In summary, the privacy protection of the lineage workflow structure needs to meet the following requirements: (1) The privacy protection method is based on a strict privacy protection model and can measure the degree of privacy security quantitatively; (2) Maintain the availability of key paths that describe the evolution of data items in the lineage workflow.

3.2 Related Definition

Definition 1. ε-differential privacy [14]: Data set D_1 and D_2 have at most one record different, range(F) is the range of the random function F. If F satisfies the following conditions, then F is said to satisfy the ε-differential privacy model:

$$\Pr[F(D_1) \in S] \leq \exp(\varepsilon) \times \Pr[F(D_2) \in S],\ S \subseteq \text{Range}(F)$$

Among them, ε is called privacy protection budget, and D_1 and D_2 are called adjacent data sets.

Definition 2. Global Sensitivity [17]: For the function f: D → Rd, the input is the data set D and the output is the d-dimensional real vector. For any adjacent data set D_1 and D_2, the global sensitivity Δf is defined as:

$$\Delta f = \max_{D_1, D_2} ||f(D_1) - f(D_2)||$$

Theorem 1 [16]. Let f: D → Rd be a query function. If method A satisfies the following conditions, then A is said to satisfy differential privacy:

$$\mathrm{A(D)} = f(\mathrm{D}) + [Lap(\frac{\Delta f}{\varepsilon})]^{\mathrm{n}}$$

Among them, Lap(Δf/ε) is a Laplace noise variable independent of each other, and the noise magnitude is proportional to Δf and inversely proportional to ε. Therefore, the greater the global sensitivity, to achieve the same level of privacy protection, the greater the amount of noise required.

Definition 3. Module and module degree d/out degree dout/in degree din: For module m, its input I = {a1, …, at}, output O = {b1, …, bn}; then module m can be abstracted as a mapping R: I → O. The degree of the module m is d = t+n, the degree of exit is dout = n, and the degree of entry is din = t.

Definition 4. adjacent modules If there is a dependency r: mi → mj between the modules mi and mj, then mj is called the adjacency module of mi, and <mi, mj> is called an adjacency pair.

From another point of view, if the input attribute of m_j is the output attribute of m_i, then m_j is the adjacency module of m_i. As shown in Fig. 1, m_2 and m_3 are adjacent modules of m_1.

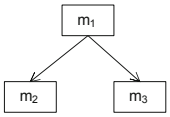


Fig. 1. Schematic diagram of the adjacent module

4 PPWP-DP

The main idea of PPWP-DP (Privacy-Preserving Workflow Publishing Method Based on Differential Privacy) is: first introduce the concept of key path and key path priority, and further propose the projection algorithm θ-Project, so that the maximum degree of the lineage workflow module is not greater than θ. Meanwhile, the priority path with higher priority is maintained more likely than the priority path with lower priority. The concept of oi-sequence is introduced to realize the feature extraction of the lineage workflow. The global sensitivity of the oi-sequence is reduced by θ-project processing, so that the oi-sequence after the disturbance satisfies the differential privacy constraint while reducing the noise scale. Finally, the perturbed oi-sequence is used to reconstruct the lineage workflow for publication.

4.1 Lineage Workflow Projection Algorithm

The dependencies between modules in the lineage workflow characterize the flow of data in the lineage workflow. If the special dependencies between the modules cannot be maintained after the privacy protection process, the workflow may be meaningless. To describe the special dependencies among modules in the lineage workflow, the concept of the key path of the lineage workflow is introduced as follows.

Definition 5. path P: For the lineage workflow W = {M, I}, the modules ms ∈ M, md ∈ M, the module md depends on the module ms through the path P = {ms → m1 m2 → … → mk → md}, and the modules in the path are not duplicated. Then P is called a path in W.

Some paths in the lineage workflow are important to the user, called the key path KP (KeyPath), and there can be multiple key paths in the lineage workflow. As shown in Fig. 2, the path P = {m_1, m_3, m_5, m_6} is the key path KP of the lineage workflow W. In this paper, unless otherwise stated, the meaning of the key path is defined by 5.

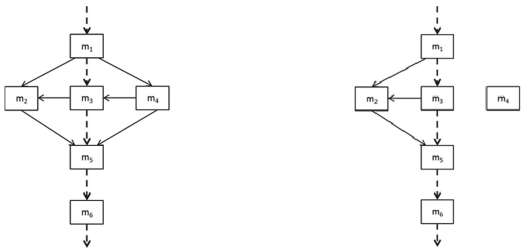
Definition 6. key path priority: KP1 and KP2 are the two key paths in the lineage workflow. If the importance of KP_1 is greater than KP_2 according to user preference, the priority of the key path KP_1 is higher than the priority of KP_2, which is recorded as $KP_2 \propto KP_1$.

Reducing the global sensitivity of the algorithm is important work. Considering the method of graph projection to reduce the global sensitivity of edge differential privacy and to ensure the accessibility of key paths with higher priority, based on this, the θ-Project algorithm is proposed. See Algorithm 1 for specific algorithm steps.

Algorithm 1 θ-Project
Input：Lineage workflow W=<M,I>, degree threshold θ, stable dependency sequence Λ_{kp} and Λ_{rest}
Output：Lineage workflow Wθ with a maximum degree less than θ

1. I^{θ}=Ø
2. d(m)=0 for each m∈M/* Set the degree of all modules in M to 0.*/
3. **for** i=(m_j， m_{j+1})∈Λ_{kp} **do**
4. **if** d(m_j)<θ and d(m_{j+1})< θ **then**
5. I^{θ}={i}∪I^{θ}; d(m_j)=d(m_j)+1; d(m_{j+1})=d(m_{j+1})+1 /* Add dependencies on key paths */
6. **end for**
7. **for** i=(m_t， m_{t+1})∈Λ_{rest} **do**
8. **if** d(m_t)<θ and d(m_{t+1})< θ **then**
9. I^{θ}={i}∪I^{θ}； d(m_t)=d(m_t)+1； d(m_{t+1})=d(m_{t+1})+1
10. **end if**
11. **end for**
12. return W^{θ}=（M，I^{θ}）

The θ-Project algorithm requires that the ordering of the DAG graph edges be stable; the so-called edge sorting is stable: the DAG graph G = (V, E) and G′ = (V′, E′) differ only by one node, but Λ(G) and Λ(G′) are consistent, that is, the relative order of the two side-by-side sorts is consistent.



(a) Lineage workflow W (b) W is processed by θ-Project to generate W'

Fig. 2. The lineage workflow W is processed before and after the θ-Project algorithm, where θ = 3, and the dotted line is the key path of the lineage workflow W.

4.2 oi-Sequence

The dk-sequence [17] is a powerful tool for describing the structural features of graphs, which describes the distribution of nodal degrees in the graph. To better describe the characteristics of the lineage workflow structure, the concept of oi-sequence is introduced as follows.

Definition 7. oi-sequence: For workflow W, its oi-sequence is represented by the set OIS: OIS = $\{ois_1, ois_2, \ldots, ois_n\}$, where ois_i is a triple: $ois_i = <d_{out}, d_{in}, count>$, d_{out} is the degree of m_i, and d_{in} is the degree of m_j, where m_j is the adjacency module of m_i, and count is the number of occurrences of the adjoining pair whose degree is d_{out} and the degree of ingress is d_{in}.

Since the lineage workflow has initial input and final output, it is convenient to extract the oi-sequence, adding "source module" and "end module" to the lineage workflow. The lineage workflow in Fig. 2 is shown in Fig. 3 after adding "source module" and "end module".

Figure 3 shows the lineage workflow after the "source module" and "end module" are added to the lineage workflow W^{θ}. The right side of the figure is its corresponding oi-sequence. W^{θ} has four sets of oi-sequences, which are denoted as set OIS(Wθ) = {<1, 1, 3>, <2, 2, 3>, <2, 1, 1>, <1, 2, 1>}. Wherein <2, 2, 3> means that there are three adjacent pairs with an outdegree of 2 and an entry degree of 2, respectively $<m_1, m_2>$ and $<m_3, m_5>$, $<m_3, m_2>$.

The oi-sequence describes the characteristics of the outflow and ingress distribution of the lineage workflow module, which can better realize the extraction of the characteristics of the lineage workflow.

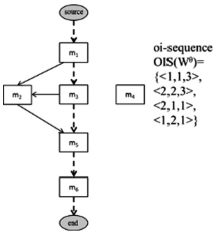


Fig. 3. The lineage workflow W^{θ} adds "source module" and "end module" and its oi-sequence.

4.3 Privacy-Preserving Workflow Publishing Method Based on Differential Privacy

After acquiring the structural feature $OIS(W^{\theta})$ of the lineage workflow W, the PPWP-DP considers using the Laplace noise perturbation $OIS(W^{\theta})$ to generate an oi-sequence that satisfies the ε-differential privacy constraint. And reconstruct the lineage workflow for release. The pseudo code of the PPWP-DP complete algorithm is given.

The complete PPWP-DP algorithm steps are given below in Sects. 4.1 and 4.2.

Algorithm 2 PPWP-DP
Input：lineage workflow W, parameter θ, privacy budget ε, time-series oi-sequence set KPS
Output：lineage workflow Wr that satisfies the ε-differential privacy model constraint

1. W^{θ}=*θ-project(W，θ)* /* Reduce the maximum degree of W */
2. Compute $OIS(W^{\theta})$ according to W^{θ} /* Generate oi-sequence according to W^{θ}*/
3. Using ε to perturb $OIS(W^{\theta})$ and acquire ε-differentially private $\tilde{OIS}(W^{\theta})$ /* Theorem 1*/
4. DDS= Ø
5. compute KPOIS for keypaths
6. Compute DDS of W^{θ} /* Calculate the DDS of W^{θ}*/
7. rebuild keypaths according to DDS， KPOIS and $\tilde{OIS}(W^{\theta})$
8. update DDS and $\tilde{OIS}(W^{\theta})$
9. Create W_r with modules but no edges， stubs using DDS
10. **For** each <d_{out}， d_{in}， count > in $\tilde{OIS}(W^{\theta})$
11. **For** i=1 to count
12. Choose any module m which has output degree d_{out} and n which has input degree d_{in}
13. **If** m does not have free output stubs
14. **Choose** u' whice has free output stubs
15. m=m';
16. **End if**
17. **If** n does not have free input stubs
18. **Choose** n' whice has free input stubs;
19. n=n';
20. **End if**
21. Add edge from m to n;
22. **End for**
23. **End for**
24. return W_r

5 Effectiveness Analysis

The lineage workflow W is processed by the θ-Project algorithm to generate W^{θ}, and the oi-sequence of W^{θ} is calculated to extract the feature of W^{θ} structure, and Laplace noise is added to the oi-sequence to satisfy the difference privacy model constraint. The global sensitivity of the oi-sequence and its proof are given below.

Theorem 2. The lineage workflow W is processed by the algorithm θ-Project to generate the lineage workflow as W^{θ}, the global sensitivity upper bound of the oi-sequence of the lineage workflow W^{θ} is 4θ−3.

Proof: e is a dependency between modules m_1 and m_2 added to the lineage workflow W^{θ}. Once e is added between m_1 and m_2, the degree of m_1 changes from d to d + 1, and the degree of m_2 changes from d' to $d' + 1$. Assume that the ingress of module m_1 is i and the outdegree of module m_2 is j. Then, after adding e, the count of $d - i + - j$ oi-sequences is increased by 1, and the count of $d - i + d' - j$ oi-sequences is decremented by one. Together with the newly added dependency, the oi-sequence change before and after adding e is 2 $(d - i + d' - j) + 1$. In the worst case, both d and d' take the maximum degree θ, then the global sensitivity of the oi-sequence is $4\theta - 2$ $(i + j) + 1$. Since the minimum value of i and j can take 1, the upper bound of $4\theta - 2$ $(i + j) + 1$ is $4\theta - 3$. The certificate is completed.

It can be known from Theorem 2 that the global sensitivity of the oi-sequence of the lineage workflow W is proportional to the maximum degree of the module in W. The θ-project algorithm reduces the maximum degree of the lineage workflow module, thus reducing the global sensitivity of the oi-sequence and reducing the Laplacian noise scale.

6 Experimental Analysis

The experimental data used two social network datasets on the Stanford Network Analysis Platform: wiki-vote (http://snap.stanford.edu/data/wiki-Vote.html), soc-sign-bitcoin-alpha (http://snap.stanford.edu/data/soc-sign-bitcoin-alpha.html). See Table 1 for details of the data set. The experimental environment is: Windows7 Ultimate, 64-bit operating system, Intel(R) Core(TM) i5-6500 3.2 GHz processor, 8 GB of installed memory.

Table 1. Experimental data set information

Data name	Nodes	Edges	Max degree	Average degree
wiki-vote	7115	103689	1167	29.15
soc-sign-bitcoinalpha	3783	24186	888	12.78

This section analyzes data availability from two perspectives: data distortion and path query reachability. The DKDP process was designed to compare with PPWP-DP. The natural logarithm LNMSE (Natural Logarithm of Mean Square Error) comparing the mean square error of the sequence after PPWP-DP and DKWP treatment was used as an experimental evaluation standard. The performance of the PPWP-DP method in critical path maintenance is verified by comparing the reachability of key path queries with the reachability of common path queries.

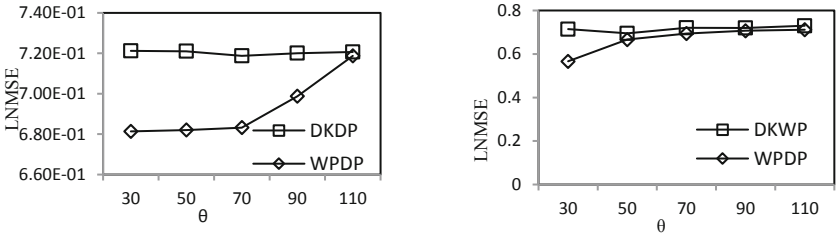
6.1 LNMSE Comparison Experiment

LNMSE is a measure used to evaluate the degree of difference between the estimator and the estimated amount. In this experiment, the difference between the true value of the oi-sequence and the post-disturbance value is evaluated using LNMSE. For DKWP, the dk- is evaluated using LNMSE. The difference between the true value of the sequence and the value after the disturbance. The calculation formula of the original sequence oi-s and the LNMSE of the perturbed sequence $\widetilde{oi-s}$ is as follows:

$$LNMSE\left(oi-s,\widetilde{oi-s}\right)=\ln\left(\frac{\sum_{i=1}^{n}\left((oi-s.count)-\left(\widetilde{oi-s}.count\right)\right)^{2}}{n}\right)$$

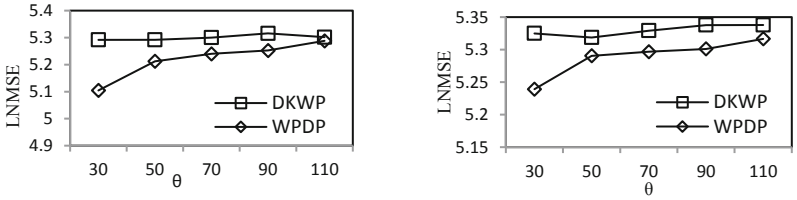
In order to ensure the reliability of the results, multiple experiments are used to average the method to avoid errors. Specifically, for the same θ value, 100 times of LNMSE values are averaged for each method as the value of the final LNMSE.

Figure 4(a) shows the experimental results on the dataset wiki-vote. Since the θ value is independent of the DKWP method, the LNMSE value of the DKWP remains around 0.72. As the θ value increases, the value of the LNMSE of the PPWP-DP method gradually increases, indicating that the data distortion is intensified. Because the global sensitivity of the oi-sequence increases as the value of θ increases, the noise scale also increases, resulting in increased data distortion. For the same θ value, the LNMSE of the PPWP-DP method is smaller than the LNMSE of the DKWP method, indicating that the DPWP method reduces the data distortion and improves the data



（a）wiki-vote （b）soc-sign-bitcoinalpha

Fig. 4. LNMSE changes with θ size when privacy budget ε = 1



（a）wiki-vote （b）soc-sign-bitcoinalpha

Fig. 5. LNMSE changes with θ size when privacy budget ε = 0.1

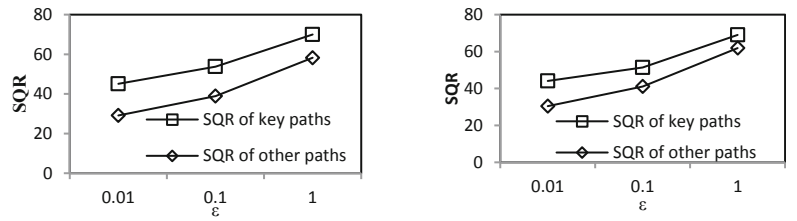
availability under the same ε value. Figure 4(b) shows the experimental results on the data set soc-sign-bitcoinalpha. Figure 5 are experimental results under the privacy budgets ε = 0.1. It can be seen that as the θ value increases, under the same conditions, the larger the value of LNMSE, the more severe the data distortion. At the same time, under the condition that the parameter θ and the privacy budget ε are the same, the LNMSE of the PPWP-DP is smaller than the LNMSE of the DKWP, indicating that the availability of the method PPWP-DP is higher than that of the DKWP.

6.2 Key Path Query

This section experimentally verifies the maintenance of key paths by the PPWP-DP. Two groups of queries were designed, 50 in each group. The first group of queries is the key path query, and the second group is the other path queries. The experiments is performed on the noise-disturbed wiki-vote dataset and the soc-sign-bitcoinalpha dataset. The success of the query means that module A and module B are connected by path p in the original workflow. If in the disturbed lineage workflow, module A and module B can still be connected by path p′, where p′ and p are the same. The definition of the query success rate SQR is given below:

$$\mathrm{SQR} = \frac{success\_count}{sum} * 100$$

Among them, success_count is number of successful queries, and sum is e total number of queries (Fig. 6).



（a）wiki-vote （b）soc-sign-bitcoinalpha

Fig. 6. SQR changes under different privacy budgets ε

7 Summary

Aiming at the problem that the existing lineage workflow structure privacy protection method can't quantitatively measure privacy security and ignore the maintenance of key path, this paper proposes a privacy protection method PPWP-DP that satisfies the ε-differential privacy constraint. Introducing the concept of key path and key path priority, the θ-project projection algorithm is designed to reduce the processing of the lineage workflow. The concept of oi-sequence is proposed to extract the features of the lineage workflow structure. The global sensitivity of the oi-sequence after adding noise is adjusted by θ-project, so that the oi-sequence after the disturbance satisfies the ε-differential while reducing the noise scale. Finally, the perturbed oi-sequence is used to reconstruct the lineage workflow for publication, which realizes the protection of workflow structure privacy and the maintenance of key path accessibility. After theoretical verification and experimental analysis, the proposed method can quantify the privacy security intensity and improve the privacy security while maximizing the accessibility of key paths.

Acknowledgment. Our work is supported by the National Natural Science Foundation of China (No. 61772131).

References

1. Simmhan, Y.L., Plale, B., et al.: A survey of data provenance in e-science. ACM SIGMOD Rec. **34**(3), 31–36 (2005)
2. Braun, U., Shinnar, A., et al.: Securing provenance. In: International Provenance & Annotation Workshop, vol. 4403, p. 752 (2008)
3. Davidson, S.B., Khanna, S., Roy, S., et al.: On provenance and privacy. In: International Conference on Database Theory, ICDT 2011, Uppsala, Sweden, 21–24 March 2011, Proceedings, DBLP, pp. 3–10 (2011)
4. Davidson, S.B., Khanna, S., Milo, T., et al.: Provenance views for module privacy (2011)
5. Wu, J., Ni, W., Zhang, S.: Generalization based privacy-preserving provenance publishing. In: Meng, X., Li, R., Wang, K., Niu, B., Wang, X., Zhao, G. (eds.) WISA 2018. LNCS, vol. 11242, pp. 287–299. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-02934-0_27
6. Davidson, S.B., Roy, S.: Provenance: privacy and security (2017)
7. Davidson, S.B., Khanna, S., Milo, T.: To show or not to show in workflow provenance. In: Tannen, V., Wong, L., Libkin, L., Fan, W., Tan, W.-C., Fourman, M. (eds.) In Search of Elegance in the Theory and Practice of Computation. LNCS, vol. 8000, pp. 217–226. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41660-6_10
8. Suriarachchi, I.: Addressing the limitations of Γ-privacy. School of Informatics and Computing, Indiana University (2014)
9. Chebotko, A., Chang, S., Lu, S., et al.: Scientific workflow provenance querying with security views 2008
10. Gil, Y., Cheung, W.K., Ratnakar, V., et al.: Privacy enforcement in data analysis workflows. Analysis (2008)
11. Gil, Y., Fritz, C.: Reasoning about the appropriate use of private data through computational workflows. In: AAAI Spring Symposium (2010)

12. Davidson, S.B., Milo, T., Roy, S.: A propagation model for provenance views of public/private workflows, pp. 165–176. Computer Science (2012)
13. Machanavajjhala, A., Gehrke, J., Kifer, D., et al.: L-diversity: privacy beyond k-anonymity. In: 22nd International Conference on Data Engineering. IEEE Computer Society (2006)
14. Dwork, C.: A firm foundation for private data analysis. Commun. ACM **54**(1), 86 (2011)
15. Dwork, C.: Differential privacy. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4052, pp. 1–12. Springer, Heidelberg (2006). https://doi.org/10.1007/11787006_1
16. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Proceedings of the 3rd Conference on Theory of Cryptography, New York, USA, pp. 265–284 (2006)
17. Mahadevan, P., Kroiukov, D., Fall, K., Vahdat, A.: Systematic topology analysis and generation using degree correlations. ACM SIGCOMM Comput. Commun. Rev. **36**(4), 135–146 (2006)
18. Tillman, B., Markopoulou, A., Butts, C.T., et al.: Construction of Directed 2K Graphs (2017)
19. Hay, M., Li, C., Miklau, G., et al.: Accurate estimation of the degree distribution of private networks. In: 2009 Ninth IEEE International Conference on Data Mining. IEEE Computer Society (2009)
20. Nissim, K., Raskhodnikova, S., Smith, A.: Smooth sensitivity and sampling in private data analysis. In: Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, STOC 2007, San Diego, California, USA, 11 June–13 June 2007, p. 75 (2007)
21. Zhang, X., Wang, M., Meng, X.: An accurate method for mining top-k frequent pattern under differential privacy. J. Comput. Res. Dev. **51**(1), 104–114 (2014)