Hindawi
Security and Communication Networks
Volume 2021, Article ID 8296417, 9 pages
https://doi.org/10.1155/2021/8296417

Research Article

Gene Sequence Clustering Based on the Profile Hidden Markov Model with Differential Identifiability

Xujie Ren, Tao Shang, Yatong Jiang, and Jianwei Liu

School of Cyber Science and Technology, Beihang University, Beijing 100083, China

Correspondence should be addressed to Tao Shang; shangtao@buaa.edu.cn

Received 17 September 2021; Revised 29 October 2021; Accepted 29 November 2021; Published 24 December 2021

Academic Editor: Ding Wang

In the era of big data, next-generation sequencing produces a large amount of genomic data. With these genetic sequence data, research in biology fields will be further advanced. However, the growth of data scale often leads to privacy issues. Even if the data is not open, it is still possible for an attacker to steal private information by a member inference attack. In this paper, we proposed a private profile hidden Markov model (PHMM) with differential identifiability for gene sequence clustering. By adding random noise into the model, the probability of identifying individuals in the database is limited. The gene sequences could be unsupervised clustered without labels according to the output scores of private PHMM. The variation of the divergence distance in the experimental results shows that the addition of noise makes the profile hidden Markov model distort to a certain extent, and the maximum divergence distance can reach 15.47 when the amount of data is small. Also, the cosine similarity comparison of the clustering model before and after adding noise shows that as the privacy parameters changes, the clustering model distorts at a low or high level, which makes it defend the member inference attack.

1. Introduction

In recent years, with the development of IoT-based gene sequencing technology, the amount of biological gene sequence data has increased rapidly [1]. The biological sequence data contains information about species evolution, genetic traits, and potential diseases in genes. With the help of biological big data and modern computing methods [2], such as machine learning technology, researchers' studies in genomics, transcriptomics, and proteomics can be further developed, providing help for disease diagnosis and treatment, drug research and development, reproductive health, and other fields.

Clustering is an unsupervised machine learning technique. By dividing the data with high similarity into one cluster and the data with low similarity into different clusters, the unlabeled data can be automatically categorized. The cluster analysis technology can be applied to the genomic data to realize the analysis of gene homology, genetic diseases, and traceability.

The concern about privacy has come at the time of the surge in data. The personal identifiable private information contained in the gene sequences is easy to be used by attackers, and the individual unique information extracted from the gene fragments will lead to the disclosure of private information [3]. Therefore, it is necessary to develop privacy protection technology for genomic data to ensure that private information will not be stolen. At present, the most popular approach adopted by some service providers is not to provide genomic data in public and open the query authority of the black-box model only to the users. However, Shokri et al. [4] proposed the construction method of a shadow model. By building a shadow data set and a shadow model, the training set of the black-box model was deduced. This attack is generic and not specific to a particular model.

At the same time, Shokri thought that differential privacy [5] (DP) is an effective defense against this kind of attack. Differentially private models are secure against membership inference attacks. By adding Laplace noise to the model, the change of a single individual in the database will not have a significant impact on the output result, which increases the difficulty of the attacker to carry out the inference attacks. Differential privacy is a definition of privacy based on strict mathematical proof. However, it defines privacy as the difference between the output of two neighbor databases,

which is inconsistent with the relevant privacy regulations, such as the U.S. HIPAA safe harbor rule [6]. To solve this problem, Lee and Clifton [7] put forward the concept of differential identifiability (DI), which defined privacy as the probability of an individual being identified by an attacker in the database, which is more consistent with people's cognition of privacy. In addition, the privacy parameters defined by differential identifiability are easier to understand by practitioners who are not major in privacy protection.

In this paper, a privacy clustering algorithm based on the profile hidden Markov model (PHMM) is proposed. In theory, some algorithms based on dynamic programming, such as Needleman–Wunsch algorithm, can also solve the exact solutions of a multiple sequence alignment. However, as the number of sequences increases, the complexity of the algorithm increases exponentially. The alignment algorithm based on the hidden Markov model can train a probability matrix for long sequences and is more suitable for the data mining analysis of large data sets. The sequences of high similarity are divided into the same cluster. To protect the private data, we add random Laplace noise based on differential identifiability into PHMM. We only use DNA sequences as the example of algorithm illustration and assume that the occurrence probability of each nucleotide is equal. The clustering process is iterative. We show the iterative allocation method of differential identifiability privacy parameters that can be used to add noise to PHMM. The experimental results show that if the privacy parameters are properly set, the proposed model is still usable after adding noise.

2. Related Work

Clustering analysis using HMM was proposed as early as 1985 by Juang and Rabiner [8], who put forward the concept of divergence distance that could effectively measure the similarity between the two HMMs and applied it to the sequence clustering analysis. Krogh et al. [9] proposed a variant of the HMM called the profile hidden Markov model that defined the hidden states as matching states, insertion states, and deletion states for multiple sequence alignment of genes and proteins. At present, such sequencing platforms as PFAM [10], HMMER [11], and SMART [12] use the hidden Markov model for multisequence alignment. Kater et al. [13] proposed a clustering robustness score that solved the problem that most clustering methods were not robust to noise. By artificially adding noise, they obtained the clusters of biologically significant cells in a single cell expression dataset. Jia et al. [14] proposed a double elite genetic clustering algorithm after an in-depth analysis of traditional clustering algorithms. This algorithm guarantees the global convergence of the population by elite strategies.

The model-based approach constructs the clustering model by the integration of a finite number of mixed submodels. Each submodel represents a class of data and is built by calculating the statistical properties of these data. Clustering based on the hidden Markov model was first proposed by Juang and Rabiner [8] and applied to protein sequence clustering by Krogh [9]. For the problem of model initialization and cluster number selection of HMM clustering, Smyth [15] provided a reference scheme. In model-based clustering, the logarithmic likelihood of the sequence and the model can be considered as the basis to measure the sequence similarity. Clustering based on HMM can be formally expressed as

$$f_k(O) = \sum_{j=1}^{k} f_j\left(O \mid \lambda_j\right), \tag{1}$$

where the parameter λ_j is obtained by the maximum likelihood estimation method in Section 4. Given the sequence O and the j^{th} model parameter λ_j, $f_j(O \mid \lambda_j)$ is the probability density of the sequence under this model. Logarithmic probability is usually used to represent the probability of the output sequence of the model to facilitate calculation and prevent underflow. The commonly used methods of calculating the sequence output probability include the forward-backward algorithm and the Viterbi algorithm [16]. Logarithmic probability can be used to describe the degree of fit between the sequence and model, i.e., the degree of homology between the sequence and gene family, and it can be used as the basis of clustering.

The concept of privacy protection can be traced back to the 1970s [17]. Traditional privacy protection methods can be divided into anonymization [18–20] and data perturbation [5, 7]. Anonymization technology is vulnerable to consistency attacks and background knowledge attacks [21], and it lacks strict mathematical proof. Differential privacy [5] is a privacy definition with a strict mathematical basis that protects privacy by limiting the influence of individuals on the output of the database. Differential privacy mechanisms mainly include the Laplace mechanism and exponential mechanism [22], and the noise size is determined by the privacy budget ε. In 2012, Lee and Clifton [7] argued that differential privacy did not pay attention to the risk of individuals being identified by their adversaries in the database, which was inconsistent with the definition of relevant laws and regulations. In addition, the parameter setting of differential privacy was a tricky issue, thus putting forward differential identifiability. Compared with differential privacy, the privacy parameter setting of differential differentiability is more intuitive and easier to be understood by relevant practitioners. Shang et al. [23] proposed the composition theorem of differential identifiability and applied it to the k-means clustering in the MapReduce framework.

3. Preliminary

3.1. Profile Hidden Markov Model. The hidden Markov model [24] is a linear serial statistical model composed of multiple observable states and corresponding hidden state nodes. Its hidden states are the unobservable underlying sequences. The transition between the hidden states is carried out according to a certain probability, and each observable state also appears according to a certain probability. In other words, HMM is a set of finite states that are transferred by a series of hidden states, which can be described indirectly by an observable sequence.

The profile hidden Markov model (PHMM) was first proposed by Krogh et al. [9] and applied to a multisequence alignment of proteins. PHMM defines the hidden state as match state, insert state, and delete state on the basis of HMM and adds a start state and an end state. Compared with the traditional HMM, PHMM is more sensitive to the specific position of the state in the observation sequence. Figure 1 shows a simple PHMM with squares representing the match state, diamonds representing the insert state, and circles representing the delete state.

Given an input gene sequence to PHMM, PHMM will match the symbols in the sequence and the insertion and deletion states and output a score. The higher the score, the more the sequence matches with the sequence family of training PHMM and the higher the gene sequence homology from the biological point of view.

The hidden Markov model has three basic problems: evaluation problem, learning problem, and decoding problem.

The decoding problem is the focus of this paper. Given that model $\lambda = (\pi, A, B)$ and observation sequence O. π, A, and B represent the initial state distribution, state transition probabilities, and observation symbol probabilities, respectively. The model outputs the most likely corresponding hidden state path and its probability value of O. The Viterbi algorithm [16] is a dynamic programming algorithm. Let $V_j^M(i)$ be the log odds of matching the sequence $x_1, \ldots, x_i$ to the optimal path of the submodel ending with state j. Similarly, $V_j^I(i)$ and $V_j^D(i)$ represent the log odds of the optimal path ending with the states I_j and D_j, respectively. The general formula of the Viterbi algorithm is as follows:

$$
\begin{aligned}
V_j^M(i) &= \log \frac{e_{M_j}(x_i)}{q_{x_i}} \\
&\quad + \max \begin{cases} V_{j-1}^M(i-1) + \log a_{M_{j-1}M_j}, \\ V_{j-1}^I(i-1) + \log a_{I_{j-1}M_j}, \\ V_{j-1}^D(i-1) + \log a_{D_{j-1}M_j}, \end{cases} \\
V_j^I(i) &= \log \frac{e_{I_j}(x_i)}{q_{x_i}} \\
&\quad + \max \begin{cases} V_j^M(i-1) + \log a_{M_jI_j}, \\ V_j^I(i-1) + \log a_{I_jI_j}, \\ V_j^D(i-1) + \log a_{D_jI_j}, \end{cases} \\
V_j^D(i) &= \max \begin{cases} V_{j-1}^M(i) + \log a_{M_{j-1}D_j}, \\ V_{j-1}^I(i) + \log a_{I_{j-1}D_j}, \\ V_{j-1}^D(i) + \log a_{D_{j-1}D_j}. \end{cases}
\end{aligned} \tag{2}
$$

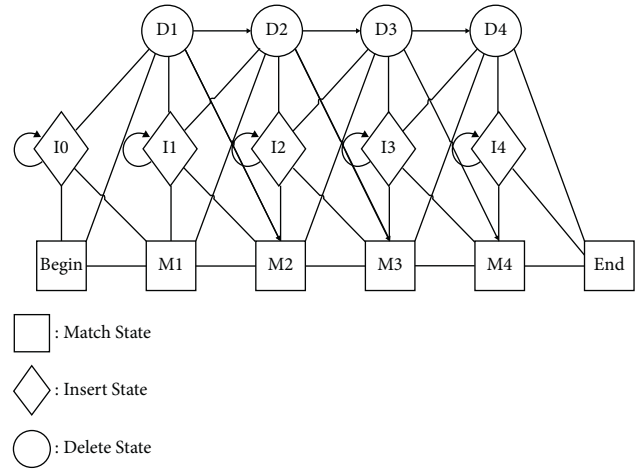

Figure 1: A simple PHMM structure.

The maximum value of the three is the log odds of the sequence corresponding to the optimal path, and the optimal path can be found after recalling.

3.2. Differential Identifiability. Differential identifiability aims to solve the problem that ε-differential privacy [5] has no clear guidelines on parameter setting. Strictly speaking, differential identifiability is another form of the implementation of differential privacy, which can provide the same privacy protection capability as differential privacy. It takes the probability ρ of individuals being identified in the database as the privacy parameter, which can be easily set by decision makers.

Definition 1. (ρ-differential identifiability).

Given a query function f, for any data set D to be queried, $\forall D' = D - t^*$ and any individual $t \in U - D'$, if

$$
\Pr\left[I(t) \in \mathcal{I}_D \mid M_f(D) = R, D'\right] \leq \rho, \tag{3}
$$

then the mechanism M satisfies ρ-differential identifiability. U represents the set composed of all possible individuals, $I(t)$ represents the identifier of individual t in the universal set U, and $\mathcal{I}_D$ represents the set of all individuals in database D, i.e., $\mathcal{I}_D = \{I(t) \mid t \in D\}$.

Definition 1 states that if an adversary possesses a neighbor data set and has the ability to query the database, the mechanism limits the probability of an attacker identifying an individual in the data set to less than or equal after he obtains the output of the database. Differential identifiability is similar to differential privacy in that they assume equal capabilities of the adversary. The difference between them is that differential privacy defines privacy as the indistinguishable output of two neighbor databases, while differential identifiability focuses on the probability of an individual being identified by an adversary in the database. In contrast, the privacy parameters of differential identifiability are more intuitive and convenient for nonprofessionals to set.

Differential identifiability can also realize privacy protection by adding random noise to the query result, i.e., $R = f(D) + Y$, Y is a random variable that obeys a certain Laplace distribution. Lee and Clifton gave a Laplace mechanism satisfying differential identifiability, i.e., $Y \sim \mathrm{Lap}(\lambda)$. Similar to differential privacy, the model is distorted after adding noise, which interferes with the attacker's inference attack and reduces the probability of a successful attack as is described in Theorem 1.

Theorem 1. *Suppose there is a random mechanism M, and for any adversary, if $\lambda = \Delta f / \ln(m-1)\rho/1-\rho$, the mechanism M satisfies ρ-differential identifiability, where $m = |\Psi| = |U| - |D'|$ and Δf is the sensitivity of the query function f.*

For data publishing in some complex situations, privacy protection mechanisms may need to be applied multiple times. On the basis of Lee and Clifton's work, Shang et al. [23] studied the composition theorem of differential identifiability. Based on the composition theorem, the differential identifiability privacy protection with mathematical proof can be provided when the adversary has the ability of multiple combinatorial queries.

Theorem 2 (Sequential Composition [23]). *Given a set of n mechanisms $M_1, \ldots, M_n$, each $M_i (i \in [1,n])$ provides ρ_i-differential identifiability. For all databases D, the sequential composition $M(D) = (M_1(D), \ldots, M_n(D))$ provides $(m^{n-1} \sum_n^{i=1} \rho_i)$-differential identifiability.*

Theorem 3 (Parallel Composition [23]). *Given a set of n mechanisms $M_1, \ldots, M_n$, each $M_i (i \in [1,n])$ provides ρ_i-differential identifiability. D_i is the arbitrary disjoint subset of the input database D. The sequence of mechanisms $M_i(D_i)$ provides $(\min \rho_i)$-differential identifiability.*

In some iterative data mining mechanisms, such as unsupervised clustering, the realization of privacy protection may require noise-adding with multiple rounds or disjoint subdata sets. Sequential composition and parallel composition, respectively, indicate that when the same data set is noised in multiple rounds and the disjoint data set is noised in separate rounds, the model obtained still meets differential identifiability.

4. Profile Hidden Markov Model with Differential Identifiability

4.1. Modeling. Adding noise to the black-box model is an effective privacy protection method to resist an inference attack [30]. The model of differential privacy makes the probability of a dataset producing an output close to its neighbor dataset [4]. Differential identifiability limits the probability that an attacker can identify a particular record in the database by perturbing the model randomly. Algorithm 1 shows the steps of constructing the profile hidden Markov model with DI. In a clustering algorithm for gene sequence in Section 5, the DI-PHMM constructed by Algorithm 1 will be used to measure the similarity between the sequences.

In PHMM, the states are defined as match, insert, and delete. We adopt the maximum likelihood estimation to estimate the parameters (transition probability and emission probability). A_{kl} represents the number of transitions from the state k to state l in the training data, and $E_k(a)$ represents the number of the emission symbol a of state k. a_{kl} represents the probability of transition from state k to state l, and $e_k(a)$ represents the probability of transmitting observation symbol a under state k. Pseudocount (for simplicity, usually set to 1) is used to add to the count result of each state or observed symbol. It is done to prevent overfitting and avoid the problem of calculating the fractions wrong when a symbol count is 0.

4.2. Privacy Parameter Allocation. The traditional differential privacy follows the mechanism that the privacy budget in each round is half of that of the last round in the iterative process. In previous work [23], according to the mapping relationship between differential identifiability and differential privacy, the privacy parameters of differential identifiability were also set to decrease by half. In essence, this method is still differential private noise and does not satisfy the differential identifiability's composition theorem.

Different from the composition theorem of differential privacy, for the sequence composition acting on the same data set, the parameter ρ of the final model is the product of the privacy parameters of each round, namely $m^{n-1} \sum_{i=1}^{n} \rho_i$, rather than the summation in differential privacy [26]. Set the expected privacy parameter ρ in advance for a finite number of iterations. For each round, the differential identifiability privacy parameter can be set to

$$\rho_i = \left(\frac{\rho}{m^{N-1}}\right)^{1/N}, \tag{4}$$

where ρ_i is the privacy parameter of iteration in round i, and N is the total time of iteration rounds. According to the sequential composition of differential identifiability, the final model meets $\rho_{\text{final}} = m^{N-1} \sum_{i=1}^{N} \rho_i = \rho$.

When the number of iteration rounds is infinite or unknown, according to sequence combination and power series convergence, it can be deduced that the privacy parameter of each round should be set as

$$\rho_i = \frac{(\rho m)^{(1/2)^i}}{m}. \tag{5}$$

4.3. Noise Addition. Counting data is generally considered a risk of being subjected to differential cryptanalysis. Accordingly, when the maximum likelihood estimation of observation symbols is carried out here, random noise Y_k is added to the technical results of each observation symbol, and k represents the corresponding hidden state. Y_k obeys the Laplace distribution with a position parameter of 0, i.e.,

Input: Training sequence data $O = (O_1, O_2, \ldots, O_n)$ and differential identifiability parameter ρ
Output: A PHMM Calculate probability of transition from state k to state l. a_{kl} = pseudocount $(A_{kl}/\sum_{l'} A_{kl'})$;
(1) //Generate privacy noise for matching state and insert state emission probabilities.
(2) for a in ('A', 'G', 'T', 'C')
(3) $Y_{ma} \sim \text{Lap}(\Delta f/\ln(m-1)\rho/1-\rho)$,
(4) $Y_{ia} \sim \text{Lap}(\Delta f/\ln(m-1)\rho/1-\rho)$;
(5) //Calculate the emission probability of state k to state a.
(6) for k in ('match', 'insert')
(7) $e_k(a)$ = pseudocount $(E_k(a) + Y_k/\sum_{a'} E_k(a') + \sum_{k'} Y_{k'})$
(8) //Validity test. Check whether the emission probability is valid. If not, go
(9) back to step 2.
(10) if $e_m(a), e_i(a) \notin [0, 1]$
(11) Regenerate privacy noise
(12) Return $a_{kl}, e_k(a)$

Algorithm 1: Profile hidden Markov model with differential identifiability (DI-PHMM).

$Y_k \sim \text{Lap}(\Delta f/\ln(m-1)\rho/1-\rho)$ (position parameter omitted). The sensitivity is usually determined by the query function. When a single individual in the database is modified, added, or deleted, the maximum value of its count changes by 1, and hence, the sensitivity $\Delta f = 1$ here. Differential identifiability defines the set U of all possible data individuals in the full set. For gene sequences, there is $|U| = 5^L$ (the space size of the bases "A, C, G, T," and placeholder "-"). Hence, $m = |U| - |D'| = 5^L - n + 1$.

The validity test of the algorithm aims to prevent the emission probability value less than 0 because of the influence of noise. When it happens, the algorithm regenerates the noise and adds noise until the new probability value is between 0 and 1.

5. Gene Sequence Clustering Algorithm Based on DI-PHMM

5.1. Clustering Algorithm. Assume there are two PHMMs λ_1 and λ_2 and one DNA sequence. Use the Viterbi algorithm on them and output the two scores s_1 and s_2. If $s_1 > s_2$, we can say that the sequence is more homologous with the sequences that constructed λ_1. Clustering for the gene sequences is an algorithm that can automatically divide the highly homologous sequences into one class. The homology analysis of the sequences is helpful for the researchers to trace the origin of genes and analyze the point of genetic diseases (such as comparing mouse genetic diseases with human genes). In Algorithm 2, a sequence clustering method based on the hidden Markov model is proposed, and the clustering model is protected by differential identifiability.

The clustering model needs to train for several rounds of iteration, and finally, it converges. The k-means is one of the most widely used clustering algorithms. The key of k-means is the measurement of the distance between the data and the cluster centers, and the data that is the closest to one center is grouped into one group. k cluster centers will update in each round of iteration, as well as data clustered according to distance metric. In general, clustering can be divided into distance-based clustering and model-based clustering according to the measurement method [27]. The classical k-means uses the Euclidean distance $d_{ij}(x_i, y_j) = \sqrt{(x_i - y_j)^T (x_i - y_j)}$ to measure the similarity between data. The smaller the distance, the greater the similarity. Hence, such data will be grouped together.

Similar to the classical clustering algorithm, the initial central submodel is selected randomly. Each sequence outputs a score for each PHMM and is divided into the corresponding cluster according to the highest score. The central submodel was updated after each round of iteration. According to whether the number of iteration rounds is preset, calculate the size of the privacy parameter of each round. Use the algorithm DI-PHMM, and the sequences within each cluster is used as input. The output model is used as the new central submodel of this cluster. Finally, the distance between the center model of this round and the previous round is calculated according to the divergence distance. If the distance is small enough, the clustering is considered to be converged. Otherwise, the iteration continues until the number of iteration rounds reaches a preset threshold or the central submodel no longer changes. Table 1 compares some gene sequence clustering models' performance.

5.2. Divergence Distance of DI-PHMM. The termination condition of clustering iteration is that the division of the clusters will not change, which is also reflected in the fact that the output score of each sequence for each submodel will not change. Juang and Rabiner [8] studied the divergence distance of the hidden Markov model, which reflects the similarity between the two HMMs. The definition of divergence distance is as follows:

$$D(\lambda_1, \lambda_2) = \lim_{T \longrightarrow \infty} \frac{1}{T} \sum_{i=1}^{T} \log \frac{f(O_i \mid \lambda_1)}{f(O_i \mid \lambda_2)}, \tag{6}$$

λ_1 and λ_2 represent the two hidden Markov models, respectively. The more similar the two models are, the smaller the divergence distance will be.

The above steps are iterated until the central submodel of the iteration no longer changes or the number of iteration rounds reaches a threshold.

Input: Number of clusters K, training sequence data $O = \{O_1, O_2, \ldots, O_n\}$, DI parameter ρ and round number of iteration N (optional)
Output: Index of the cluster to which the sequence belongs $C = \{C_1, C_2, \ldots, C_n\}$ where $C_n = 1, 2, \ldots, K$
(1) $r++$
(2) for i in n
(3) for j in k
(4) //Calculate the score of the sequence for each PHMM
(5) $d_j^i = \text{viterbi}(O_i, \lambda_j^r)$
(6) //Divide the sequence into the corresponding cluster according to the highest score
(7) $C_i = \arg\max_j (d_j^i)$
(8) for k in K
(9) if ($N!$ = NULL):
(10) $\rho_r = (\rho/m^{N-1})^{1/N}$
(11) else
(12) $\rho_r = (\rho m)^{1/2^r}/m$ //The privacy parameter is assigned according to whether the number of iteration rounds is fixed.
(13) //Construct a new cluster center sub-model
(14) $\lambda_k^r = \text{DI} - \text{PHMM}(O_k, L, \rho_r)$
(15) //The degree of change of the model from the last iteration (divergence distance)
(16) $D_k = D(\lambda_k^r, \lambda_k^{r-1})$

Algorithm 2: Gene sequence clustering algorithm based on DI-PHMM (DI-GSCA).

Table 1: Gene sequence clustering models' performance.

Model	Availability[1] (%)	Efficiency	Security
DKHC [28]	98.26	Slow	×
DEGCA [14]	96.7	Slow	×
CD-HIT [29]	95	Fast	×
GeneRage [30]	95	Fast	×
DI-GSCA	CS[2]: 0.77	Slow	✓

[1]The availability data were the highest value in their respective references.
[2]CS: cosine similarity.

6. Experimental Results

In this section, we will demonstrate the influence of privacy protection on clustering performance by comparing it with the output results of the unnoised model. The dataset is *Mus musculus* Immunoglobulin Lambda Chain Complex (IGL) on chromosome 16 and *Homo sapiens* chromosome 1 genomic from the National Center for Biotechnology Information. A total of 10,000 sequences with a length of 70 bp were intercepted from them. The experimental results were the average values of 10 experiments.

6.1. Distortion of DI-PHMM. Based on the description of the algorithm of DI-PHMM, the divergence distance between the native and noised PHMM models is given to show the effect of noising on the PHMM model. The privacy parameter of differential identifiability is set to nine values of 0.1 to 0.9. Generally speaking, a smaller privacy parameter indicates that the adversary has a lower probability of identifying the original data set. The noise that needs to be added is higher, and the distortion of the model is higher. In this section, the training data sets are randomly divided into 10,000, 5000, 2000, and 1000 sequences to build the PHMM model, respectively. Generally, the larger the data set, the more robust it is to noise. Therefore, when the data set size is large, the influence of noise processing on the PHMM model is small. The experimental results are shown in Figure 2.

Because of the randomness of the adding noise, the divergence distance of the PHMM model after adding the noise may fluctuate to a certain extent. As a whole, PHMM constructed with the size of 10,000 sequences has the smallest divergence distance after adding the noise, ranging from 0.01 to 8.56. PHMM constructed from a dataset of only 1000 sequences showed a large degree of distortion after noising. When the privacy parameter was 0.1, the divergence distance reached a maximum of 15.47.

Obviously, with the increase of the privacy parameter, the difference between the noise-added model and the normal model decreases, indicating that the noise is indeed reduced. At the same time, larger data sets show stronger robustness to noise. It seems that the fluctuation range of the divergence distance of $n = 5000$ is higher than that of $n = 2000$. It is because of the randomness of noise.

6.2. Performance of Clustering Model. The output result of the clustering algorithm described in the gene sequence clustering algorithm based on DI-PHMM is a vector of the class corresponding to the sequence data set. Cosine similarity is adopted in this section to measure the similarity between the clustering results. The cosine similarity is evaluated by calculating the cosine of the angle between the two vectors. Given two vectors A and B, the cosine similarity between them can be expressed as $\text{similarity} = A \cdot B/\|A\|\|B\| = \sum_{i=1}^n A_i \times B_i / \sqrt{\sum_{i=1}^n (A_i)^2}\sqrt{\sum_{i=1}^n (B_i)^2}$. The value of cosine similarity ranges from 0 to 1. When the cosine similarity is 0, it means that the two vectors are perpendicular and the similarity is 0. When the cosine similarity is 1, the two vectors are equal.

By comparing the cosine similarity between the clustering results constructed by the PHMM model without the noise and privacy clustering results, the influence of adding

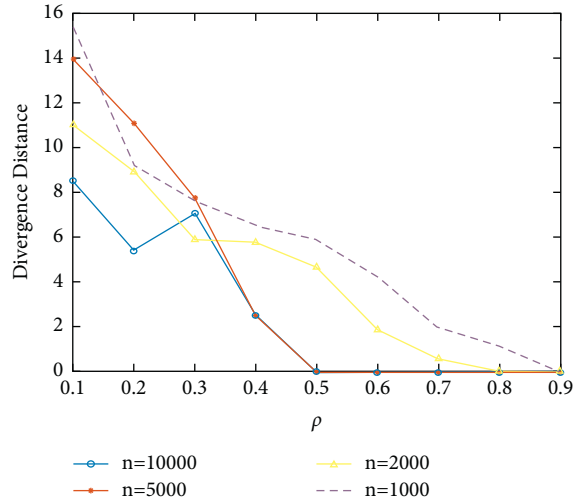

Figure 2: Divergence distance of PHMM before and after noise addition.

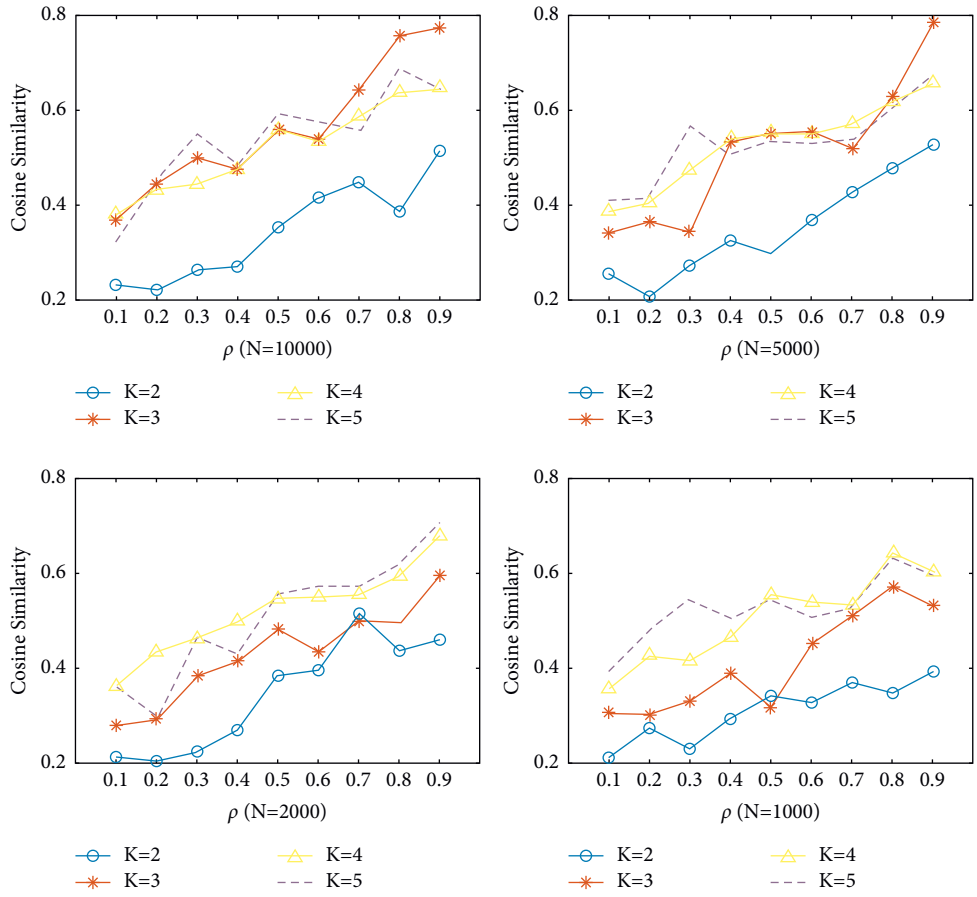

Figure 3: Cosine similarity of clustering model.

differential identifiability noise on the clustering performance is demonstrated. In the experiment, the number of clusters was set as $K = 1, 2, 3, 4$, and the experimental results are shown in Figure 3.

The experimental results show that with an increase in the privacy parameters, the privacy noise decreases gradually, the degree of privacy protection decreases, and the clustering performance improves accordingly. When we set the appropriate number of clustering and privacy parameters, the clustering results after adding the noise can still maintain an acceptable usability. For example, when $N = 10,000$, the number of clustering is 3, and the privacy

parameter is 0.9, the clustering performance reaches the highest, and the cosine similarity is 0.77 compared with the normal model.

7. Conclusion

On the basis of the classical gene sequence clustering algorithm, we added the privacy noise, satisfying the differential identifiability into the clustering model so that the model can resist the member inference attack and also provide a valuable privacy protection method for researchers related to biological information. In this paper, an iterative allocation scheme of differential identifiability privacy parameters was presented to realize privacy protection in complex models. The experimental results show that the clustering utility of the model with differential identifiability noise is reduced, however, the model availability can still be maintained at a high level after adjusting the privacy parameters.

Future research can focus on the basic theory of differential identifiability, how to balance privacy and model performance, and applications in more scenarios.

Data Availability

The gene sequence data used to support the findings of this study is cited from the NCBI repository (https://www.ncbi.nlm.nih.gov/).

Conflicts of Interest

The authors declare that no conflicts of interest exist.

Acknowledgments

This project was supported by the National Key Research and Development Program of China (No. 2016YFC1000307) and the National Natural Science Foundation of China (Nos. 61971021 and 61571024).

References

[1] Z. Alansari, N. B. Anuar, A. Kamsin, S. Soomro, and M. R. Belgaum, *Progress in Advanced Computing and Intelligent Engineering*, Springer, Singapore, 2019.

[2] P. Mamoshina, A. Vieira, E. Putin, and A. Zhavoronkov, "Applications of deep learning in biomedicine," *Molecular Pharmaceutics*, vol. 13, no. 5, pp. 1445–1454, 2016.

[3] Y. Erlich and A. Narayanan, "Routes for breaching and protecting genetic privacy," *Nature Reviews Genetics*, vol. 15, no. 6, pp. 409–421, 2014.

[4] R. Shokri, M. Stronati, and C. Song, "Membership inference attacks against machine learning models," in *Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, IEEE, San Jose, CA, USA, 2017.

[5] C. Dwork, *Differential Privacy: A Survey of Results. International Conference on Theory and Applications of Models of Computation*, Springer, Berlin, Heidelberg, Getmany, 2008.

[6] Office for Civil Rights H H S, "Standards for privacy of individually identifiable health information," *Final rule, Federal Register*, vol. 67, no. 157, pp. 53181–53273, 2002.

[7] J. Lee and C. Clifton, "Differential identifiability," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1041–1049, ACM, New York, NY, USA, August 2012.

[8] B.-H. Juang and L. R. Rabiner, "A probabilistic distance measure for hidden Markov models," *AT&T Technical Journal*, vol. 64, no. 2, pp. 391–408, 1985.

[9] A. Krogh, M. Brown, I. S. Mian, K. Sjölander, and D. Haussler, "Hidden Markov models in computational biology," *Journal of Molecular Biology*, vol. 235, no. 5, pp. 1501–1531, 1994.

[10] R. D. Finn, A. Bateman, J. Clements et al., "Pfam: the protein families database," *Nucleic Acids Research*, vol. 42, no. D1, pp. D222–D230, 2014.

[11] R. D. Finn, J. Clements, W. Arndt et al., "HMMER web server: 2015 update," *Nucleic Acids Research*, vol. 43, no. W1, pp. W30–W38, 2015.

[12] I. Letunic, T. Doerks, and P. Bork, "SMART: recent updates, new developments and status in 2015," *Nucleic Acids Research*, vol. 43, no. D1, pp. D257–D260, 2015.

[13] I. Kanter, P. Dalerba, and T. Kalisky, "A cluster robustness score for identifying cell subpopulations in single cell gene expression datasets from heterogeneous tissues and tumors," *Bioinformatics*, vol. 35, no. 6, pp. 962–971, 2019.

[14] R. Y. Jia, F. B. Song, and S. W. Tang, "Genetic clustering algorithm with double elite genetic strategy," *Journal of Chinese Computer Systems*, vol. 41, no. 7, pp. 1375–1380, 2020.

[15] P. Smyth, "Clustering sequences with hidden Markov models," *Advances in Neural Information Processing Systems*, pp. 648–654, 1997.

[16] R. Durbin, S. R. Eddy, A. Krogh, and M Graeme, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press, Cambridge, UK, 1998.

[17] T. Dalenius, "Towards a methodology for statistical disclosure control," *Statistik Tidskrift*, vol. 15, no. 429–444, pp. 2–1, 1977.

[18] P. Samarati and L. Sweeney, *Protecting Privacy when Disclosing Information: K-Anonymity and its Enforcement through Generalization and Suppression*, Carnegie Mellon University, Pittsburgh, PA, USA, 1998.

[19] A. Machanavajjhala, J. Gehrke, D. Kifer, and M Venkitasubramaniam, "l-diversity: privacy beyond K-anonymity," in *Proceedings of the International Conference on Data Engineering*, pp. 24–35, IEEE, Atlanta, GA, USA, April 2006.

[20] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: privacy beyond k-anonymity and l-diversity," in *Proceedings of the International Conference on Data Engineering*, pp. 106–115, IEEE, Istanbul, Turkey, April 2007.

[21] P. M. V. Kumar and M. Karthikeyan, "L-diversity on k-anonymity with external database for improving privacy preserving data publishing," *International Journal of Computer Applications*, vol. 54, no. 14, 2012.

[22] F. MeSherry and K. Talwar, "Mechanism design via differential privacy," in *Proceedings of the 48th Annual IEEE Symposium on Foundation of Computer Science*, pp. 94–103, IEEE, Providence, RI, USA, October 2007.

[23] T. Shang, Z. Zhao, X. Ren, and J. Liu, "Differential identifiability clustering algorithms for big data analysis," *Science China Information Sciences*, vol. 64, no. 5, Article ID 152101, 2021.

[24] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[25] H. Ding, Y. Tian, C. Peng, Y. Zhang, and S. Xiang, "Inference attacks on genomic privacy with an improved HMM and an

RCNN model for unrelated individuals," *Information Sciences*, vol. 512, pp. 207–218, 2020.

[26] X. J. Ren, T. Shang, and J. W. Liu, "An iterative allocation method of privacy parameter for differential identifiability," *Journal of Cryptologic Research*, vol. 8, no. 4, pp. 582–590, 2021.

[27] S. Zhong and J. Ghosh, "A unified framework for model-based clustering," *Journal of Machine Learning Research*, vol. 4, pp. 1001–1037, 2003.

[28] J. H. Wu, J. W. Luo, and Y. Wang, "A double k-mean clustering algorithm for sequential gene data based on the hidden Markov model," *Computer Engineering & Science*, vol. 29, no. 3, pp. 54–56, 2007.

[29] W. Li and A. Godzik, "Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences," *Bioinformatics*, vol. 22, no. 13, pp. 1658-1659, 2006.

[30] A. J. Enright and C. A. Ouzounis, "GeneRAGE: a robust algorithm for sequence clustering and domain detection," *Bioinformatics*, vol. 16, no. 5, pp. 451–457, 2000.