Clustering-based Multidimensional Sequence Data Anonymization

Morvarid Sehatkar
University of Ottawa
Ottawa, ON, Canada
msehatkar@uottawa.ca

Stan Matwin
[1]Dalhousie University
Halifax, NS, Canada
[2]Institute for Computer Science of the Polish Academy of Science
Warsaw, Poland
stan@cs.dal.ca

ABSTRACT
Sequence data mining has many interesting applications in a large number of domains including finance, medicine, and business. However, Sequence data often contains sensitive information about individuals and improper release and usage of this data may lead to privacy violation. In this paper, we study the privacy issues in publishing multidimensional sequence data. We propose an anonymization algorithm, using hierarchical clustering and sequence alignment techniques, which is capable of preventing both identity disclosure and sensitive information inference. The empirical results show that our approach can effectively preserve data utility as much as possible, while preserving privacy.

Categories and Subject Descriptors
H.2.7 [**Database Management**]: Database Administration--*Security, integrity, and protection*

General Terms
Algorithms, Performance, Experimentation, Security

Keywords
Data anonymization, privacy, multidimensional sequence data, longitudinal data, clustering, *k*-anonymity

1. INTRODUCTION
Recent advances in information technology have enabled public organizations and corporations to collect and store huge amounts of individuals' data in data repositories. Such data are powerful sources of information about an individual's life such as interests, activities, and finances. Corporations can employ data mining techniques to extract useful knowledge from individuals' data and exploit this knowledge to improve their strategic decision making, enhance business performance, and improve services. As a result, the demand for collecting and sharing data has been rapidly increased. Among various types of individuals' data, event sequence data mining has many interesting applications in a large number of domains. Sequence data mining enables us to discover behaviour patterns of individuals through temporal activities. Such knowledge is precious for planning, detecting behavioral changes, and commercial purposes. For instance, longitudinal medical records of patients can be used to analyze patients' reactions to a new drug or to support a diagnosis. However, despite all benefits of analyzing event sequence data, this data often contain sensitive information and may violate privacy of individuals if published. In event sequence data, every event may have a number of attributes that act as *quasi-identifiers* (*QIs*). Due to temporal correlation among the events of each sequence, in addition to the values of *QIs* within an event, any combination of *QIs* values across events along with the temporal information about these values might lead to privacy breach. For example, consider Table 1 containing information of multiple visits of patients in a hospital over the last five years. Every visit corresponds to a multidimensional event and the ordered list of these events represents one sequence. Each event has 5 attributes, including admission year (*AdmYr*), ZIP code, number of days since the first visit in each year (*DSFC*), and the length of stay in the hospital (*LOS*), which all act as *QIs*, as well as one sensitive attribute diagnosis. An adversary with some background knowledge about visits of a target individual is able to launch two types of privacy attacks: *identity disclosure* and *attribute disclosure*. For instance, if the adversary knows that Bob had a visit in 2009 and he has been living in ZIP code 56230 from 2010, she can uniquely identify Bob's record, #6, and consequently conclude that Bob has *HIV*. In case of attribute disclosure, if the adversary knows that Bob had a visit in 2007 and *later* in 2011 he was hospitalized for 3 days, then she can conclude that Bob has *HIV* since both matching records to her knowledge, #8 and #9, have *HIV* in one of their visits.

A common practice for releasing individuals' data without violating privacy is *data anonymization*. Data anonymization techniques aim to modify data such that no sensitive information about individuals can be disclosed from published data while data distortion is minimized to ensure usefulness of data in practice. In order to effectively anonymize multidimensional sequence data, to prevent both identity disclosure and attribute disclosure attacks, temporal correlation among the events of each record should be considered in anonymization process, and it should be guaranteed that no combination of values of *QI*s within an event and across events of any record leads to privacy breach. In the past years, several anonymization algorithms were proposed to protect privacy when publishing different types of data [2]. However, none of these methods are applicable to anonymize a multidimensional sequence dataset, like the data in Figure 1 (a). Recently, a few methods have been designed to anonymize longitudinal health data which is a case of event sequence data [1][6][7]. However, authors in [1] and [7] only focused on privacy protection against identity disclosure. Moreover, in the longitudinal data, studied in [7], each record contains a sequence of (*ICD*, *Age*) pairs as well as a DNA sequence where ICD represents the code of the diagnosis made for a patient and Age is the patient's age at the time of diagnosis. Considering such data, background knowledge of an adversary in this method is modeled as any combination of (ICD, Age) pairs. Obviously, this method

Table 1 Patient data of a hospital

PID	VID	AdmYr	ZIP	DSFC	LOS	Disease
1	1	2009	56117	0	3	Hepatitis
2	1	2007	56103	0	2	Infection
3	1	2008	56942	0	1	Fever
3	2	2010	56942	0	30	Infection
4	1	2008	56107	0	2	Fever
4	2	2010	56107	0	35	Flu
5	1	2009	56117	0	3	Fever
6	1	2009	56103	0	3	Flu
6	2	2009	56103	10	1	Fever
6	3	2010	56230	0	2	HIV
7	1	2008	56072	0	2	Flu
8	1	2007	56361	0	30	Hepatitis
8	2	2011	56107	0	3	HIV
9	1	2007	56230	0	35	Flu
9	2	2011	56107	0	3	HIV
10	1	2009	56072	0	2	Flu
10	2	2009	56103	13	35	Fever
10	3	2010	56043	0	30	Infection

Table 2 Anonymized patient data satisfying (2, 0.5)-*privacy*

PID	VID	AdmYr	ZIP	DSFC	LOS	Disease
1	1	2009	56117	0	3	Hepatitis
2	1	[2007:2008]	56***	0	2	Infection
3	1	[2007:2008]	56***	0	[0:12)	Fever
3	2	[2009:2012]	56***	0	[0:12)	Infection
4	1	[2007:2008]	56***	0	[0:12)	Fever
4	2	[2009:2012]	56107	0	[0:12)	Flu
5	1	2009	56117	0	3	Fever
6	1	2009	56***	0	[0:1)	Flu
6	2	2009	56103	[1:2)	[0:12)	Fever
6	3	2010	56***	0	[0:12)	HIV
7	1	[2007:2008]	56***	0	2	Flu
8	1	[2007:2008]	56***	0	[0:12)	Hepatitis
8	2	[2009:2012]	56107	0	[0:12)	HIV
9	1	[2007:2008]	56***	0	[0:12)	Flu
9	2	[2009:2012]	56***	0	[0:12)	HIV
10	1	2009	56***	0	[0:1)	Flu
10	2	2009	56103	[1:2)	[0:12)	Fever
10	3	2010	56***	0	[0:12)	Infection

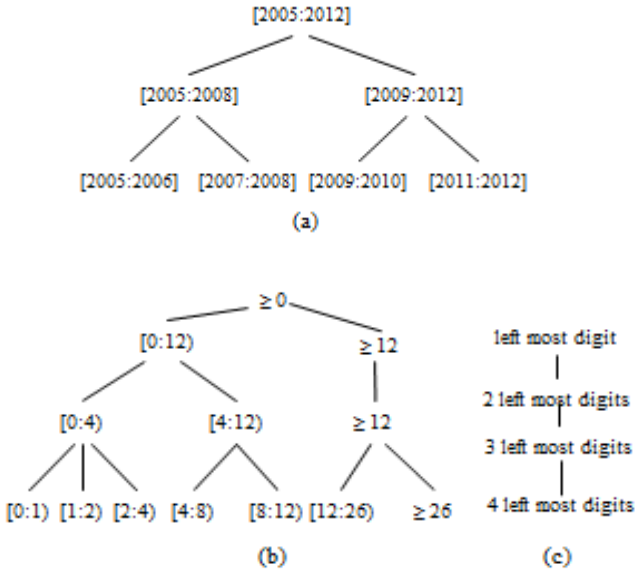

Figure 1 Generalization hierarchy for (a) AdmYr (b) DSFC and LOS in terms of number of weeks (c) ZIP

is limited to two *QIs* and fails to consider the multidimensionality of events in our problem. In [1] it is assumed that adversaries would not have any information about co-occurrence of values of quasi-identifiers in one event as well as the order of events of a target individual. As a result this work fails to model all potential background knowledge of adversaries. The proposed method in [6] prevents both identity disclosure and attributes disclosure; however knowledge of adversaries is assumed to be limited to at most p values of quasi-identifiers. Although this assumption decreases information loss, determining the appropriate value of p is not trivial. As a result the adequate level of privacy protection may not be achieved. In this paper, we define a new privacy model called (k,c)-*privacy* to anonymize multidimensional sequence data to prevent identity disclosure and attribute disclosure. This privacy model ensures that every combination of values of *QI*s within an event and across events of any sequence is shared by at least k sequences, and the probability of inferring any sensitive value is at most c. We achieve (k, c)-privacy by presenting an anonymization algorithm based on *hierarchical agglomerative clustering* [4] and *sequence alignment* [5] techniques. We assume that the purpose of data publication is unknown and so our algorithm anonymizes data by minimizing overall data distortion. Table 2 shows an anonymized version of the data in Figure 1(a) satisfying (2, 0.5)-*privacy* using generalization hierarchies in Figure 1.

2. PROBLEM DEFINITION

In this section we present the framework which forms the basis of our anonymization methodology. Specifically, we describe the privacy model and the utility measure.

2.1 Privacy Model

Suppose a data holder wants to share its multidimensional sequence data for public use. Let $A = \{A_1, A_2, \dots, A_n\}$ be a set of attributes and $\Delta = \{\Delta_1, \Delta_2, \dots, \Delta_n\}$ be the corresponding attribute domains. Each A_z is either a categorical or a numerical attribute. Also assume there is one sensitive attribute ψ with the domain values $\Delta_\psi = \{s_1, \dots, s_l\}$. A multidimensional sequence dataset D is a collection of records of the form (*SID*, *S*), where *SID* is a unique id for every individual and S is an ordered list of multidimensional events, denoted by $S = \langle e_1, e_2, \dots, e_m \rangle$. Each event e has the form $(EID, a_1, a_2, \dots, a_n, s)$ where *EID* is the event's id, a_z is a domain value of A_z, $a_z \in \Delta_z$, and s is a value of the sensitive attribute ψ, $s \in \Delta_\psi$. Events of every sequence S are ordered with respect to temporal information of one of the attributes $A_z \in \{A_1, A_2, \dots, A_n\}$. We refer to the value of the z^{th} *QI* attribute of the j^{th} event of the sequence S_i by $e_{ij}(z)$ and the value of the sensitive attribute ψ in the j^{th} event of the sequence S_i is denoted by $e_{ij}(\psi)$. A subset of attributes $\{A_1, A_2, \dots, A_n\}$ is assumed to be publicly available, so they act as quasi-identifiers, $QIs \subseteq \{A_1, A_2, \dots, A_n\}$. The values of the sensitive attribute are naturally private. We assume an adversary who knows that the record of a target individual exists in a released multidimensional sequence dataset. She also has some background knowledge about the sequential events of a target individual, i.e. the values of some *QIs* as well as the order of these values in some of the events of an individual's sequence. Armed with this knowledge, the adversary seeks to find some *matching* records to her background knowledge in the released data. If the number of such records is not "sufficiently" large or the percentage of sequences among these records containing a common sensitive value σ is high, the adversary may infer some sensitive information about the individual. Since adversaries'

knowledge is assumed to be in the form of any combination of *QIs'* values, the worst-case scenario would be an adversary knowing the values of *all QIs'* in *all* events of a target individual. Therefore, to protect privacy of individuals the privacy model should ensure that every sequence in the released data is linked to a sufficiently large number of other sequences and the percentage of sequences with the same sensitive value in every group of indistinguishable sequences is not too high. However, the latter case may not need to be satisfied for every value of the sensitive attribute. More precisely, if some values of the sensitive attribute have less degree of sensitivity and do not need to be kept private, then we do not need to be worried about these values being too frequent in a group. For example, in the context of publishing medical data, it might be allowed to disclose the value "flu" for the sensitive attribute disease. To effectively handle these cases, we define a set $\Omega \subseteq \psi$, called *highly-sensitive* set, which contains those values of the sensitive attribute ψ which have a high degree of sensitivity. In the presence of this set, our privacy model must ensure that the frequency of sequences which have at least one of the values in Ω in some of their events is not too high in any group of indistinguishable sequences. This brings us to the following definition.

DEFINITION 1 ((*k*, *c*)-privacy). Given anonymity threshold $k \geq 2$, and confidence threshold $c \in (0,1]$, a multidimensional sequence dataset *D* satisfies (*k*, *c*)-privacy if *i*) each sequence in *D* is indistinguishable from at least *k-1* other sequences with respect to any combination of *QIs* and *ii*) the probability of inferring any high sensitive value in any group of indistinguishable sequences is at most *c*.

2.2 Information Loss

We employ generalization and suppression on the values of *QIs* to modify data and form clusters. This anonymization process incurs information loss because some original values of *QIs* in every sequence are either replaced with less specific values or are totally removed. In order to preserve data utility for data mining tasks, we should ensure that anonymization cost is minimized. We consider the scenario where the data analysis task is unknown at the time of data publication. So, our goal is to anonymize a multidimensional sequence data to satisfy (*k*,*c*)-privacy while preserving data utility as much as possible. Let D^* be an anonymization of the multidimensional sequence data *D*. D^* corresponds to a set of clusters $C=\{C_1, C_2, ..., C_p\}$ which is a clustering of sequences in *D*. All sequences in a given cluster C_j are anonymized together. We define the amount of information loss incurred by anonymizing *D* to D^* as

$$IL(D, D^*) = \frac{1}{|D|} \sum_{j=1}^{p} IL(C_j) \quad (1)$$

where $IL(C_j)$ is the information loss of the cluster C_j, which is defined as the sum of information loss of anonymizing every sequence *S* in C_j:

$$IL(C) = \sum_{i=1}^{|C|} IL(S_i, S_i^*) \quad (2)$$

where $|C|$ is the number of sequences in the cluster *C*, and $IL(S, S^*)$ is the information loss of anonymizing the sequence *S* to the sequence S^*.

Each sequence is anonymized by generalizing or suppressing some of the *QIs'* values in some of its events. So, we define information loss of a sequence based on the information loss of its events. Let *H* be generalization hierarchy of the attribute *A*. We use the *Loss Metric* (*LM*) measure [3] to capture the amount of information loss incurred by generalizing the value *a* of the attribute *A* to one of its ancestors $\hat{a}$, with respect to *H:*

$$IL(a, \hat{a}) = \frac{|\mathcal{L}(\hat{a})| - |\mathcal{L}(a)|}{|\Delta_A|} \quad (3)$$

where $|\mathcal{L}(x)|$ is the number of leaves in the subtree rooted at *x*.

The information loss of each event *e* is then defined as

$$IL(e, e^*) = \sum_{n=1}^{|QI|} IL(e(n), e^*(n)) \quad (4)$$

where $e^*$is the ancestor of the event *e*, $e(n)$ is the value of n^{th} *QI* of the event *e* and $e^*(n)$ is its corresponding value in the event e^*.

Hence, the information loss incurred by anonymizing each sequence is as follows:

$$IL(S, S^*) = \sum_{m=1}^{|S|} IL(e_m, e_m^*) \quad (5)$$

3. ANONYMIZATION ALGORITHM

We propose a bottom-up anonymization algorithm based on hierarchical agglomerative clustering. The general idea is to anonymize data by starting with the trivial clustering that consists of singleton clusters and then keep merging the two closest clusters, until all clusters satisfy privacy constraints based on (*k*,*c*)-privacy model. A key factor in any clustering algorithm is the distance measure. In order to minimize the overall data distortion due to anonymization, we define the distance between two given clusters as the change in information loss when we merge the clusters:

$$dist(X, Y) = IL(X \cup Y) - IL(X) - IL(Y) \quad (6)$$

where $IL(X \cup Y)$ is the information loss of the merged cluster, and $IL(X)$ and $IL(Y)$ are information loss of clusters *X* and *Y* before merge, respectively.

We assume that every cluster has a representative sequence which is the result of anonymizing all contained sequences. The distance between two clusters is calculated based on the information loss of anonymizing their representatives, and the clusters with the smallest distance are chosen to be merged. In general, two representative sequences have different number of events. So, the anonymization of these sequences can be seen as the problem of finding a matching between the events of these sequences, using generalization and suppression, such that the anonymization cost is minimized. The following definition expresses the information loss of a merged cluster based on the information loss of anonymizing representatives of two clusters which are being merged to their best matching.

DEFINITION 2. Let $\ddot{\mathrm{X}}$ and $\ddot{\mathrm{Y}}$ be representative sequences of clusters *X* and *Y* and $\mathrm{M_{XY}}$ be their best matching. Then the information loss of the merged cluster $X \cup Y$ is define as

$$\mathrm{IL(X \cup Y)} = |\mathrm{X}| \cdot \mathrm{IL}\left(\ddot{\mathrm{X}}, \mathrm{M_{X,Y}}\right) + |\mathrm{Y}| \cdot \mathrm{IL}\left(\ddot{\mathrm{Y}}, \mathrm{M_{X,Y}}\right) \quad (7)$$

where $\mathrm{IL}\left(\ddot{\mathrm{X}}, \mathrm{M_{X,Y}}\right)$ and $\mathrm{IL}\left(\ddot{\mathrm{Y}}, \mathrm{M_{X,Y}}\right)$ are information loss of anonymizing representative sequences $\ddot{\mathrm{X}}$ and $\ddot{\mathrm{Y}}$ to their best matching sequence $\mathrm{M_{XY}}$.

Finding the best matching between two sequences is a *sequence alignment* problem. The basic principle underlying sequence alignment methods is to measure the effort it takes, in terms of specific operations, to make sequences equal. One of the most common approaches for sequence alignment is *dynamic programming*. Dynamic programming is an advanced algorithmic technique that solves optimization problems from the bottom up by finding optimal solutions to subproblems. Inspired by [7], we

employ dynamic programming to align representatives of clusters with the goal of minimizing anonymization cost. The operations which we use to align (anonymize) two sequences are generalization and suppression. If two values of the attribute $q \in QI$ are identical, their generalization is equal to the values themselves; otherwise both values are replaced with their *lowest common ancestor* (*LCA*) which is the lowest node in the generalization hierarchy H_A that is an ancestor of both *v* and *w*. Given sequences $\ddot{X} = \{x_1, x_2, \dots, x_l\}$ and $\ddot{Y} = \{y_1, y_2, \dots, y_t\}$ as the representatives of two clusters, the optimal alignment of these two sequences is the alignment which incurs minimum information loss considering all *QIs*. We have three cases for aligning $\ddot{X}$ and $\ddot{Y}$: *1*) aligning $\{x_1, x_2, \dots, x_{l-1}\}$ and $\{y_1, y_2, \dots, y_{t-1}\}$, and generalizing x_l and y_t, which means replacing every *QI* value in x_l and its corresponding *QI* value in y_t with their *LCA*, *2*) aligning $\{x_1, x_2, \dots, x_{l-1}\}$ and $\{y_1, y_2, \dots, y_t\}$, and suppressing x_l, *3*) aligning $\{x_1, x_2, \dots, x_l\}$ and $\{y_1, y_2, \dots, y_{t-1}\}$, and suppressing y_t.

For every $q \in QI$ we create a score matrix to store the cost of all sub-problems for aligning two one-dimensional sequences resulted from projecting sequences $\ddot{X}$ and $\ddot{Y}$ on *q*. Each of these solutions have an anonymization cost and our objective is to find the best alignment with minimum information loss. The cost of each solution is calculated as the sum of its cost for every $q \in QI$. Besides the score matrices, we assume a *move* matrix *M* where each cell *M*[*i, j*] contains the operation which is chosen to align the sequence prefix $x_1, x_2, \dots, x_i$ and the sequence prefix $y_1, y_2, \dots, y_j$. To build the sequence $M_{X,Y}$ which is the result of best alignment of sequences $\ddot{X}$ and $\ddot{Y}$, we do a "traceback" on matrix *M* from cell *M*[*l*+1,*t*+1] to cell *M*[0,0].

Our clustering algorithm *clustering based multidimensional sequence data anonymizer* (*CBMSA*) is based on agglomerative hierarchical clustering. We start with the trivial case of singleton clusters and iteratively merge two closest clusters which are determined by applying our multidimensional sequence alignment algorithm. Once, a cluster satisfies (*k,c*)-privacy, it will not be merged anymore. In order to reduce information loss, our preference is to merge two closest clusters which do not violate the confidence constraint of (*k,c*)-privacy model when being merged. When we merge two clusters *X* and *Y* with representative sequences $\ddot{X}$ and $\ddot{Y}$, all sequences in clusters *X* and *Y* are anonymized with respect to $M_{X,Y}$. This means that those events which are suppressed in $\ddot{X}$ and $\ddot{Y}$ based on the best alignment result are suppressed in all sequences in clusters *X* and *Y*, respectively. The remaining events of every sequence are then replaced with their corresponding events in $M_{X,Y}$. However, since we only apply generalization on *QI* values, the values of sensitive attribute in these events remain unchanged. Since our goal is to build clusters which satisfy (*k*,*c*)-privacy, for every cluster we should check if it contains at least *k* sequences and if the frequency of sequences which have at least one event with a high sensitive value is not greater than *c*. When we merge two clusters *X* and *Y*, the size of the new cluster is simply the sum of the number of sequences in *X* and *Y*. For the diversity check, we should count the number of sequences which have at least one event with a high sensitive value. In order to efficiently count these sequences, for every cluster we use a data structure, denoted by *HighSensList*, to keep track of these sequences. When we merge two clusters *X* and *Y*, the number of sequences with high sensitive value in *X* or *Y* may decrease. This is due to the fact that some events may be suppressed in sequences of cluster *X* or *Y*. If the events which are suppressed in a sequence are the only ones which contain high sensitive value, then this sequence will not contain any high sensitive value after applying suppression. So it should be removed from *HighSensList* of the cluster where it is consist of. So, after applying anonymization on sequences of clusters *X* and *Y*, we first update *HighSensList* of these clusters and then merge two *HighSensLists* to build the *HighSensList* of the new merged cluster. We keep merging clusters till no more than one cluster left. If the remained cluster does not satisfy privacy constraints, we remove all sequences contained in this cluster from data.

4. EXPERIMENTS

In this section, our goal is to evaluate the performance of our proposed anonymization algorithm in terms of information loss calculated based on Equation 1 as well as scalability by varying the anonymity threshold *k* and the confidence threshold *c*. We developed a data generator to generate synthetic multidimensional sequence data inspired from the Heritage Health Prize (*HHP*) claims data set[1]. In [2], anonymization of the *HHP* claims dataset is studied in order to prevent identity disclosure attacks. Authors identified 6 *QIs* for the claims dataset among which we selected two attributes *days since first claim in each year* (*DSFC*) and *length of stay* (*LOS*). We did not include attributes *diagnosis* and *CPTCode*[2] since these attributes are sensitive attributes in the framework of our study. Also, we disregarded attributes *place of service* and *specialty* due to not accessing to their possible original values in the *HHP* data. Instead, for each claim we included two other *QI* attributes, i.e. *ZIP code* of patients due to the fact that this information is often updated at every visit and the year in which a claim took place (*AdmYr*). We generated multiple synthetic datasets by varying the number of sequences, average number of events per sequence (3,5, and 10), and number of *QI*s per event (2, 3, and 4). For every set of data characteristics, we generated 10 datasets, evaluated their performance in terms of information loss, and took the average of information loss of 10 datasets in each set. We implemented our algorithms in Java and conducted experiments on a 1.80 GHz Intel core i5 PC with 8 GB RAM. To illustrate the benefits of our proposed multidimensional sequence alignment method, we also developed a baseline algorithm which does not use dynamic programming. If two sequences *X* and *Y* are of the same size, the baseline algorithm simply applies generalization to every event of two sequences. Otherwise, it first randomly suppresses $n = abs(|X| - |Y|)$ events in the longer sequence and then generalizes every remaining events in two sequences. In the first set of experiments, we evaluate the information loss *IL* by varying the value of the anonymity threshold *k* while keeping the confidence threshold *c* fixed. Figure 2 shows the *IL* for two datasets of size 1000 and 10000 with the average number of events 5 and three *QIs* with anonymity threshold $5 \leq k \leq 50$ and a fixed confidence threshold c = 0.7. As *k* increases, *IL* increases for both algorithms. This illustrates the trade-off between privacy and data utility. In other words, as *k* increases, higher level of privacy protection is required to keep the probability of re-identifying a target individual or inferring sensitive information about a target individual fairly low. Therefore, more data distortion occurs. Also, comparing the information loss of our anonymization algorithm based on dynamic programming with the baseline algorithm depicts the benefits of our method. In the second set of experiments we change *c* from 0.5 to 0.9 for the fix value *k* = 5. This setting allows us to measure the performance of our anonymization

[1] http://www.heritagehealthprize.com/c/hhp/data

[2] Current Procedural Terminology Code

algorithm against attribute disclosure for a fixed k. The resulting information loss of two algorithms for two datasets is shown in Figure 3. In general, *IL* decreases as c increases due to a less restrictive privacy requirement. Similar trends were observed between CBMSA and Baseline for the other datasets. The results are omitted for brevity. In Figure 4, we show the time performance of our algorithm for two datasets with $5 \leq k \leq 50$ and a fixed confidence threshold c=0.7. For a small dataset with total number of events about 4500, for every value of k, the total runtime of our algorithm is less than 30 *sec*. For a large dataset, with total number of events about 47000, the execution time of our algorithm for different values of k is between 2200 *sec* and 2700 *sec*. The run time of baseline algorithm for both datasets is very fast. This indicates that our algorithm spends a large amount of its running time on multidimensional sequence alignment based on dynamic programming. Even though the run time of our algorithm for large datasets is fairly high, we believe it is still acceptable in practice due to the fact that most anonymization tasks are off-line procedures.

5. CONCLUSIONS AND FUTURE WORKS

In this paper, we proposed an anonymization algorithm for multidimensional sequence data using sequence alignment techniques and agglomerative hierarchical clustering. To the best of our knowledge, this is the first work for multidimensional sequence data anonymization which prevents both identity disclosure and attribute disclosure without making any assumption about the knowledge of the adversary. Our experimental results on synthetic data show the effectiveness of our proposed algorithm for anonymizing multidimensional sequence data. Our future work includes the following. In this work we assumed that the goal of data publication is unknown and we anonymized data by minimizing data distortion for general data analysis purposes. In our future work, we will consider the case of publishing data for a specific data mining task such as classification. This requires employing an appropriate anonymization cost measure to capture the utility of our algorithm for data mining tasks. Moreover, in this paper, we studied the simplest case of a single sensitive attribute in every event of a sequence. An extension of our work would be the case of multiple sensitive attributes. Also, we will run experiments on real datasets to further investigate the effectiveness of our proposed algorithm.

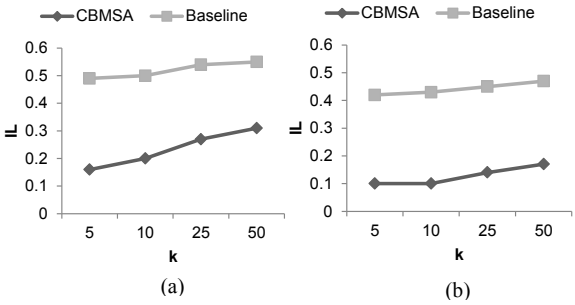

Figure 2. Information loss for (a) Data_1000_5_3 (b) Data_10000_5_3 with c = 0.7

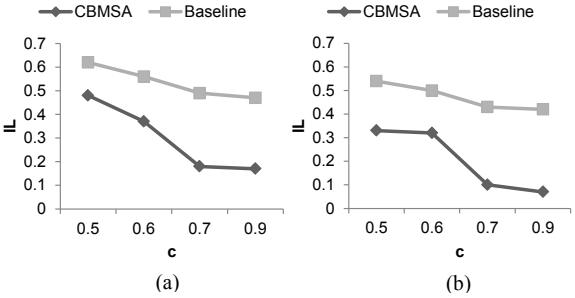

Figure 3. Information loss for (a) Data_1000_5_3 (b) Data_10000_5_3 with k = 5

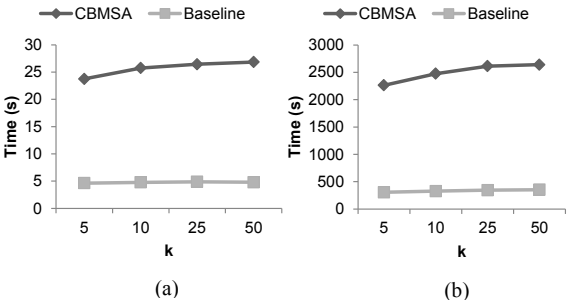

Figure 4. Execution time for (a) Data_1000_5_3 (b) Data 10000 5 3 with c = 0.7

Acknowledgment

The authors would like to thank Dr. Khaled El Emam (EHIL lab) for his inspiration with this research. The early stages of this work were partially supported by grants of CIHR and NSERC.

6. REFERENCES

[1] El Emam, K, Arbuckle, L., Koru,G., Gaudette, L., Neri, E., Rose, S., Howard, J., and Gluck, J., 2012. De-Identification Methods for Open Health Data: The Case of the Heritage Health Prize Claims Data Set. *In Journal of Medical Internet Research*, 14:1, DOI:10.2196/jmir.2001, 2012.

[2] Fung, B.C.M, Wang, K, Chen, R. and Yu, P.S. 2010. Privacy-preserving data publishing: A survey of recent developments. *ACM Comput. Surv. 42, 4, Article 14* (June 2010), 53 pages

[3] Iyengar, V.S. 2002. Transforming data to satisfy privacy constraints. In KDD '02:Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, 2002, pp. 279-288.

[4] Kaufman L and Rousseeuw, P. J., 1990. Finding Groups in Data: An Introduction to Cluster Analysis. *John Wiley* 1990

[5] Needleman, S. B. and Wunsch, C. D. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Molecular Biol.*, vol. 48, no. 3, pp. 443–453, 1970.

[6] Sehatkar, M. and Matwin S., 2013. HALT: Hybrid Anonymization of Longitudinal Transactions, Eleventh Annual International Conference on Privacy, Security and Trust (PST), Tarragona, Spain.

[7] Tamersoy, A., Loukides, G., Nergiz, M. E., Saygin, Y. and Malin, B. 2012. Anonymization of Longitudinal Electronic Medical Records. *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, pp. 413-423, 2012