Privacy-preserving Sequential Pattern Mining in distributed EHRs for Predicting Cardiovascular Disease

Eric W. Lee, MS[1], Li Xiong, PhD[1], Vicki Stover Hertzberg, PhD[2], Roy L. Simpson, RN, DNP[2], Joyce C. Ho, PhD[1]
[1]**Department of Computer Science, Emory University, Atlanta, GA**
[2]**Nell Hodgson Woodruff School of Nursing, Emory University, Atlanta, GA**

Abstract

From electronic health records (EHRs), the relationship between patients' conditions, treatments, and outcomes can be discovered and used in various healthcare research tasks such as risk prediction. In practice, EHRs can be stored in one or more data warehouses, and mining from distributed data sources becomes challenging. Another challenge arises from privacy laws because patient data cannot be used without some patient privacy guarantees. Thus, in this paper, we propose a privacy-preserving framework using sequential pattern mining in distributed data sources. Our framework extracts patterns from each source and shares patterns with other sources to discover discriminative and representative patterns that can be used for risk prediction while preserving privacy. We demonstrate our framework using a case study of predicting Cardiovascular Disease in patients with type 2 diabetes and show the effectiveness of our framework with several sources and by applying differential privacy mechanisms.

Introduction

The rapid growth of electronic health records (EHRs) provides rich information about patients' conditions, treatments, and outcomes[1]. EHRs have been widely used in various healthcare researches such as risk predictions[2,3] and phenotyping[4,5]. Yet, an often overlooked aspect of mining EHRs is the temporal nature of the data. As the data contains the previous and current status of patients, EHRs can be viewed as a sequential database chronologically ordered by date and time. Thus, sequential pattern mining (SPM) can be applied to EHRs to discover interesting, useful, and unexpected patterns that can be used by a predictive model to forecast the patient's future disease status from the current and previous patient conditions.

Existing works performing SPM of EHRs have demonstrated its potential predictive power. For example, Wright *et al*.[6] mined sequential patterns of diabetes medication prescriptions to predict the next medication to be prescribed. Ghosh *et al*.[7] proposed to apply SPM to stream bed monitors in ICUs for predicting acute hypotension in critical care patients. Lee and Ho[2] used a sequence of diagnosis codes in clinical records to predict chronic heart failure using SPM. However, the existing SPM-based methods assume all the EHRs are stored in a central repository or database. In practice, each healthcare system may have one or more clinical data warehouses to store all the patient data. Thus, one of the key challenges towards developing robust models that generalize across multiple systems is to mine data that are distributed across multiple locations or sources.

While mining distributed data itself can be challenging, a further complication arises from privacy laws. A key challenge related to large-scale analysis of EHRs is that the privacy of human subjects should be protected. Therefore, patient data in its original form cannot be transmitted without privacy guarantees that protect against some privacy attacks such as learning information about individuals from data release. To alleviate this issue, differential privacy[8] (DP) has emerged as one of the strongest privacy guarantees for statistical data release of sources such as EHRs.

In this paper, we propose a privacy-preserving SPM-based framework for mining EHRs across multiple sources. Unlike existing SPM methods that work only for EHRs stored at a central (single) source, we propose to extract discriminative or representative patterns separately at each source, share the patterns in a DP-preserving manner to a centralized location, and use the patterns for future risk prediction. A major benefit of our framework is that it can guarantee patient privacy for each source separately and still achieve approximately the same overall predictive performance of the model as the central model. We demonstrate our framework using a case study of predicting cardiovascular disease in patients with type 2 diabetes. The experimental results illustrate the effectiveness and flexibility of our framework using different DP mechanisms with several sources.

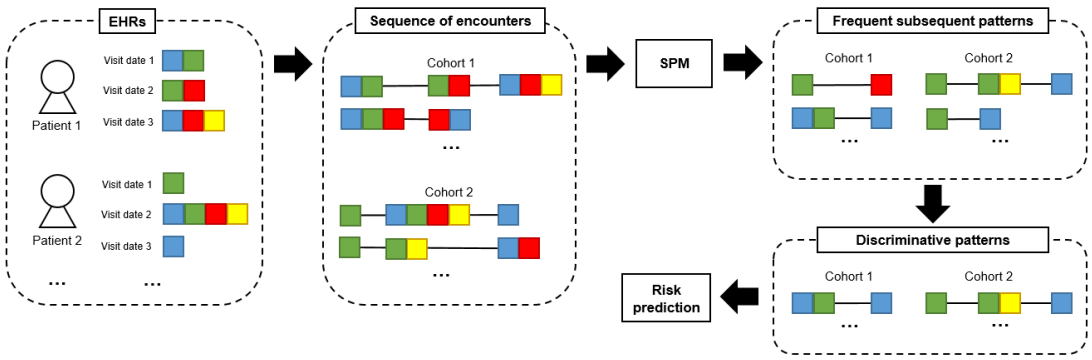


Figure 1: An illustration of the SPM-based framework in risk prediction. Each colored box in the figure denotes a single diagnosis code. There exist only one sequence of encounters for each patient. Note that for both frequent subsequent patterns and discriminative patterns, not all the patterns are shown in the figure.

A Case Study of Cardiovascular Disease in Patients with Type 2 Diabetes. Approximately 8.2% of the US population suffered from diabetes in 2018[9]. Moreover, diabetes can lead to other health complications and often results in heavy economic burden[10]. One common complication for patients with diabetes is cardiovascular disease (CVD) which refers to a number of heart-related conditions including heart disease, stroke, and heart failure. CVD incurs heavy health and economic burdens with a projected medical cost of $358 billion in 2015[11]. There is also a strong correlation between diabetes and CVD. The mortality risk of CVD among people with diabetes is high; 65% (age 65 or older) die because of heart disease and 16% die of stroke[12]. As the healthcare expenditure and resources are high in patients with diabetes and CVD, early intervention in CVD patients can lead to favorable health outcomes[13]. Thus, we demonstrate our method to predict whether a patient with diabetes will develop CVD in the future.

Background

Sequential Pattern Mining. In the field of data mining, pattern mining is broadly used to discover interesting, useful, and unexpected patterns in the database[14]. When the ordering of the events is important, SPM is proposed as a prominent solution. The goal of SPM is to find the set of all frequent subsequent patterns in the sequence database that satisfies a user-specified threshold θ. Here, θ indicates the frequency of the subsequent pattern (also known as the support count) that appears in the database.

SPM can be applied to EHRs as it contains sequences of medical evidences and actions. As an example, in each patient's visit (or encounter), ICD-9 or ICD-10 diagnosis codes are recorded. A sequence of encounters can be represented by listing each patient's encounters in chronological order by the visit dates. Once, the sequence of encounters is constructed, any SPM algorithm can be applied to extract frequent subsequent patterns.

SPM-based Framework in Risk Prediction. To mine useful patterns that can be used for risk prediction, it is often helpful to find representative patterns that distinguish patients exposed to the disease (i.e., cases) from patients not exposed to the disease (i.e., controls). Lee and Ho[2] refer to these representative patterns as *discriminative patterns*. In other words, a discriminative pattern is one that appears in one cohort but not in the other. These patterns can then be used as a feature representation for risk prediction. Figure 1 illustrates the process.

To obtain discriminative patterns, Lee and Ho[2] proposed the application of SPM to extract all frequent subsequent patterns from the sequence of encounters of each cohort that satisfies the user-specified support count. A lower support count is used to extract more patterns to improve patient representation. However, the extracted patterns may be common patterns that exist in the other cohort. To discard these common patterns, patterns existing in the other cohort are filtered out to obtain *discriminative patterns*. However, this may be too restrictive as many of the discovered patterns exist in both cohorts[2]. Thus, a threshold, τ, was proposed to allow some patterns to exist in the other cohort but require a higher support count that satisfies τ. Suppose the frequent subsequent pattern p in cohort c_1 has a support count of 10. If $\tau = 2$ and p has a support count of 5 in the other cohort c_2, then p is a discriminative pattern for cohort c_1. This process is done for both cohorts to obtain discriminative patterns for each cohort.

Differential Privacy. To protect the privacy of a human subject, Differential Privacy[8] (DP) was proposed. Under DP, the main goal is to learn useful information from the EHRs while nothing is learned about the patient. In other words, although any patient's record is arbitrarily changed, the output of an algorithm should be approximately the same. The formal definition of DP is as follows.

Definition 1. (ϵ-Differential privacy) *A privacy algorithm K satisfies ϵ-differential privacy if and only if for all neighboring databases D and D' differing on at most one record and for any possible output $S \subseteq Range(K)$,*

$$Pr[K(D) \subseteq S] \leq exp(\epsilon) \times Pr[K(D') \subseteq S] \tag{1}$$

From the Equation (1), ϵ is a privacy budget which is a metric to determine how strict the privacy is. A smaller value of ϵoffers better privacy protection. Common DP mechanisms to achieve ϵ differential privacy is the Laplace mechanism[8] and Exponential mechanism[15]. Laplace mechanism achieves ϵ differential privacy by adding a random noise sampled from the Laplace distribution to a statistical measure, and the Exponential mechanism uses exponential distribution for adding noise.

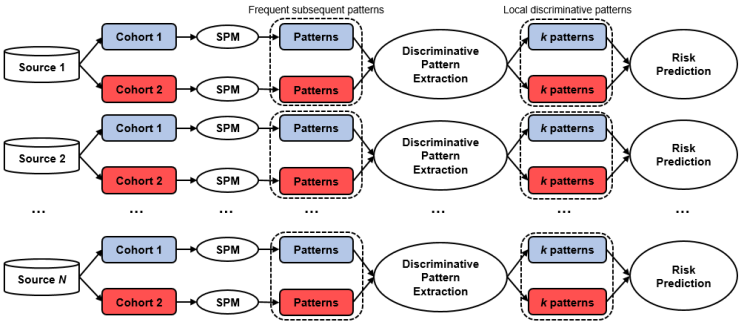
Differential Privacy in Multiple Distributed Sources. Many DP mechanisms are proposed for a traditional centralized source where all data is located in a single source[8,15,16]. There are cases when the data is distributed across multiple sources (decentralized sources). For such cases, federated learning is used to train the model across multiple decentralized sources without the sharing of raw data[17]. And for sensitive data such as EHRs, privacy is an important issue when using federated learning. Thus, many frameworks are proposed for federated learning with DP[18–20] which collaboratively train the model while preserving privacy. For example, Truex *et al*.[19] proposed to combine DP and secure multiparty computation (SMC) in a federated learning system to address the risk inference during the model learning process, and Choudhury *et al*.[20] proposed to apply DP in distributed EHRs for prediction of adverse drug reaction and mortality rate. However, for the SPM-based federated learning framework, we propose to apply DP mechanisms to support counts of subsequent patterns in multiple decentralized data sources for privacy guarantee.

Methods

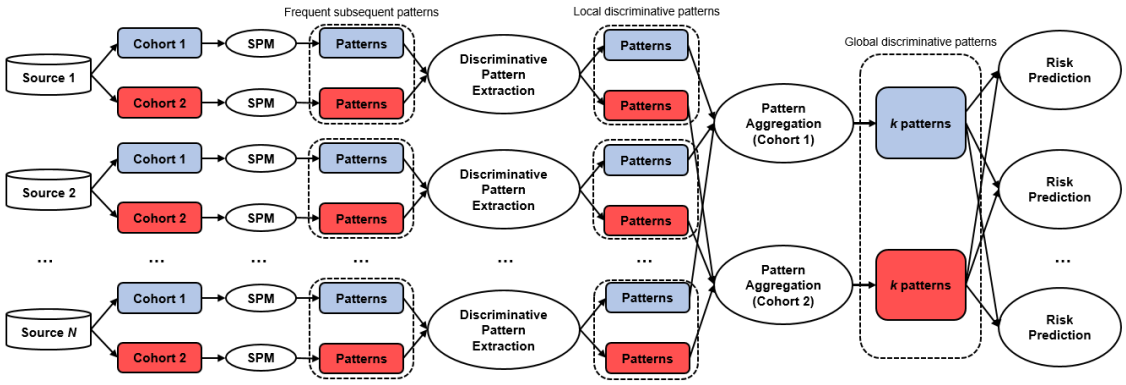
In this section, we propose our SPM-based framework for distributed EHRs. We first show a framework that trains the model individually for each data source (no aggregation framework), then we propose a framework that collaboratively trains the model by sharing the patterns from each data source while preserving the privacy (aggregation framework).

SPM-based Framework with No Aggregation. For the SPM-based framework with no aggregation, we introduce a federated learning-based technique that does not share any patient information across different data sources. Given distributed data sources, we apply the SPM-based framework and train the risk prediction classifier for each source separately. It is important to note that we call the obtained discriminative patterns as "local" discriminative patterns as it is obtained within a single data source. Once the local discriminative patterns are extracted from each cohort, we construct a feature representation based on these patterns. From the obtained patterns, we use the top k discriminative patterns based on their support counts from each cohort and use it as our feature representation for risk prediction. We use the existence of the pattern as a feature for the machine learning classification model. Suppose the top 2 discriminative patterns are extracted, p_1 and p_2 from cohort c_1, and p_3 and p_4 for cohort c_2. We construct a feature representation as $[p_1, p_2, p_3, p_4]$, and set the feature to be the existence of the pattern. For example, if one patient contains patterns p_1 and p_4, then the feature of this patient will be $[1, 0, 0, 1]$. This representation is depicted in Figure 2(a). As illustrated in the figure, all processing is performed "locally" for each data source, and none of the pattern information is shared between other data sources. In addition, each classifier is trained locally using only the patients from the same data source. One limitation of this framework is that discriminative patterns from other data sources are not used which can be more important. The disease prevalence can vary depending on location and populations, thus, to learn a stronger classifier, it may be necessary to use information from other data sources.

Privacy-preserving SPM-based Framework with Aggregation. To alleviate the limitation, we propose a SPM-based framework with aggregation that uses a centralized server to aggregate local discriminative patterns from all data sources and selects the top k global discriminative patterns which are used to train each classifier for the data source. Similar to the SPM-based framework with no aggregation, we apply SPM to each data source, extract frequent subsequent patterns, and obtain local discriminative patterns for each cohort. However, unlike the SPM-based framework



(a) SPM-based framework with no aggregation



(b) SPM-based framework with aggregation

Figure 2: A framework overview of SPM-based frameworks without and with aggregation. The blue box denotes patterns associated with cohort 1, and the red box denotes patterns associated with cohort 2.

with no aggregation, we use "global" discriminative patterns as a feature representation instead of "local" patterns. We aggregate all local discriminative patterns from each data source into a single list. In other words, there will be two lists of global discriminative patterns for each cohort (*i.e.*, one for the case and one for control). When aggregating the patterns from a centralized location for each cohort, we take the union of patterns and do the summation of the support counts of each pattern. From this aggregated list of patterns, we select the top k discriminative patterns and use them as the feature representation. These top k patterns are then shared back to the original source to train classifiers of each source. It is important to note that although all the local classifiers share the same global discriminative patterns, only the patients from one data source are used to train each classifier. The illustration of the framework is shown in Figure 2(b).

Unlike the SPM-based framework with no aggregation, this framework aggregates local discriminative patterns from all data sources into a centralized server. As patient information is shared across data sources, privacy protection becomes necessary. Therefore, once the frequent subsequent patterns are extracted from each cohort, we apply a DP mechanism such as Laplace mechanism[8] or Exponential mechanism[15] to the support counts of each pattern. The noisy support counts will be used to extract local discriminative patterns from each data source and also used when aggregating local patterns from all data sources into a centralized server. This may lead each cohort to extract a different set of local discriminative patterns compared to the framework without DP because it will make some patterns that were not frequent to become frequent and vice versa under a specific threshold. In this way, although the framework is using global discriminative patterns, privacy can be preserved by not sharing the exact patient support count information from the data source. Later, we demonstrate that the results of applying DP mechanisms will only cause a marginal decrease in predictive performance compared to the results without applying DP.

Experiment Settings

Dataset. We use Project NeLL™(Nursing electronic Learning Laboratory), a database that contains de-identified electronic health records from more than 1 million patients seen at Emory Healthcare from 2012 to 2018. It contains over 8 million unique records, including structured text (*e.g.*, lab values) and unstructured text (e.g., clinical notes, radiology reports). Patients with type 2 diabetes are identified using the ICD-9 code of '250.*' or the ICD-10 code of 'E11.*'. Note that only patients with the admitting, discharge, or final diagnosis of type 2 diabetes are used, thus ensuring that these patients are more likely to suffer from the disease. These patients who then develop cardiovascular disease (CVD) are identified using the ICD-9 codes of '428.*' or '414.*' or the ICD-10 codes of 'I50.*' or 'I25.*' (those relating to chronic heart failure and coronary heart disease). Only patients who developed CVD after diabetes are considered (i.e., any patient that had pre-existing CVD prior to diabetes is not considered in our cohort) and only the ICD-9 codes before the CVD is recorded are used. Also, any patients who have only one visit are excluded as there aren't sufficient events to model.

From the patient records, we use demographic variables such as gender, age, and race. Each encounter of patients is listed chronologically based on their visit dates. For the purpose of this study, we focus only on the discharge diagnosis codes associated with each encounter. Instead of fine-grained ICD-9 codes, Clinical Classifications Software (CCS) codes[21], a categorization scheme for the International Classification of Diseases, is used to group ICD-9 into broader categories to yield better interpretability of the patterns. For each visit, there can be multiple CCS codes. Moreover, each patient has a different number of visits with the length of the sequence of encounters varying from 2 to 546 with an average sequence length of 12.65.

Case-Control Cohort Study. Given the imbalanced ratio of CVD patients to non-CVD patients, we designed a case-control study to identify useful sequences of events. Without this process, the extracted patterns will be dominated by non-CVD patients. Thus, we matched non-CVD patients to CVD patients in a ratio of $4:1$. Patients are grouped based on the age when diabetes was first diagnosed as well as their ethnicity. The top 4 nearest patients based on Euclidean distance from the non-CVD patients are then matched to the CVD patients such that each non-CVD patient is from the same race and of a similar age as a CVD patient. Therefore there will be at most 4 non-CVD patients for every CVD patient. The resulting dataset contains 2,112 patients with CVD and 10,464 non-CVD patients, representing 34% of patients from the original dataset.

Experimental Design. To construct the sequences of encounters, we only consider encounters after the date of diabetes were developed. We also adopt the FuzzyGap sequence representation[2] to construct the sequence of encounters. This representation is constructed by setting a user-specified boundary range – encounters within the boundary between two intervals will be added to both intervals. We set the interval to 1 month, where encounters within the same month are recorded into a single encounter and 12 days for the boundary range. We note that FuzzyGap also captures gap-sensitive frequent patterns such as $[\{\text{CCS codes}\}, \{\}, \{\text{CCS codes}\}]$ which allows an empty encounter between two encounters each with a set of CCS codes. After processing the sequence of encounters to be in FuzzyGap sequence representation, we end up having 2,112 CVD patients and 7,998 non-CVD patients after excluding patients who have only one interval.

To evaluate the efficiency of our framework, we split the dataset into several partitions (or data sources). We explore 4 different partition settings: 1, 2, 4, and 8. Partition setting with 1 represents the single data source. For simplicity, we will denote these settings as *n=1*, *n=2*, *n=4*, and *n=8* for the partition settings 1, 2, 4, and 8, respectively. For each partition setting (*e.g.*, *n=4*), we evaluate 5 different random partitions. And for every random partition, there is a train-test split with a ratio of 70% and 30% respectively, and this is done 3 times by randomly selecting patients for train-test splitting. In total, we are running 15 experiments for each partition settings. For every train-test split, we have 7,078 patients in the train set and 3,032 patients in the test set.

While there are several fast and memory-efficient SPM algorithms such as FAST[22], CM-SPADE[23], and CloFast[24], our preliminary experiments using these algorithms implemented in the SPMF library[25] ran out of memory or only could obtain patterns with high support count (on a machine with 100GB of RAM). Thus, we discovered patterns by performing a sequential pairwise comparison between two patients as used in FuzzyGap[2]. As Lee and Ho[2] discussed previously, the predictive power is similar between other SPM algorithms and pairwise comparison.

To select the best top k, we first evaluate various top k settings (from 40 to 700) in the *n=1* setting without any DP mechanisms. Note that top k means k discriminative patterns from each class, hence, $2 \times k$ patterns are used as the feature representation. For extracting the discriminative patterns, we use the filtering threshold, $\tau = 2$, which allows some patterns to exist in the other cohort. Once we select the best top k from *n=1*, we use the same top k throughout the remaining experiments.

To evaluate the impact of DP mechanisms on our SPM-based framework with aggregation, we apply three DP mechanisms, Laplace mechanism[8], Exponential mechanism[15] and SVT[16]. For consistency, the privacy budget, $\epsilon = 0.1$ is used as 0.1 is a small value for ϵ and it provides strong privacy protection.

Evaluation Metrics. We evaluate the risk prediction task using the F1 score and area under the receiver operating curve (AUC). In addition to evaluation using the predictive task, we also evaluate our framework based on the recoverability of the discriminative patterns with a single data source. We use precision and recall in the information retrieval context which is defined as below.

$$precision = \frac{|relevant \cap retrieved|}{|retrieved|} \tag{2}$$

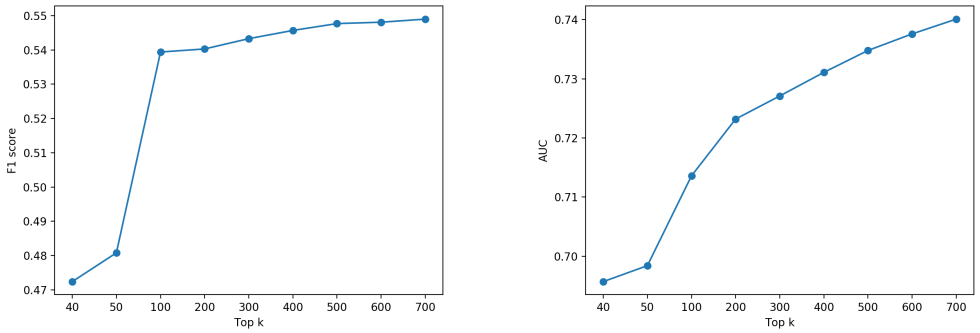
$$recall = \frac{|relevant \cap retrieved|}{|relevant|} \tag{3}$$

For both metrics, relevant refers to the discriminative patterns extracted from the single data source, and retrieved denotes the discriminative patterns extracted from the distributed sources. Only the top k discriminative patterns from each class are compared and those patterns are used as the feature representation.

Empirical Results

In this section, we use the term "*no aggregation*" for the SPM-based framework with no aggregation and no privacy, "*no privacy*" for SPM-based framework with aggregation and no privacy, and "*Laplace*", "*Exponential*", and "*SVT*" for privacy-preserving SPM-based framework with aggregation using different DP mechanisms. Note that the reported results are the average of all 3 trials.

Selecting Top k. When applying the *no aggregation* framework to *n=1* setting, many patterns are extracted. For example, in our case-control study, we discovered 924 discriminative patterns for CVD while 27,360 discriminative patterns are discovered for non-CVD patterns. Since the patterns are used for risk prediction, direct usage of all patterns may not yield desirable results due to the potential overfitting of the downstream predictive models. Thus, we use the top k discriminative patterns from each class, resulting in $2 * k$ patterns used as the feature representation.

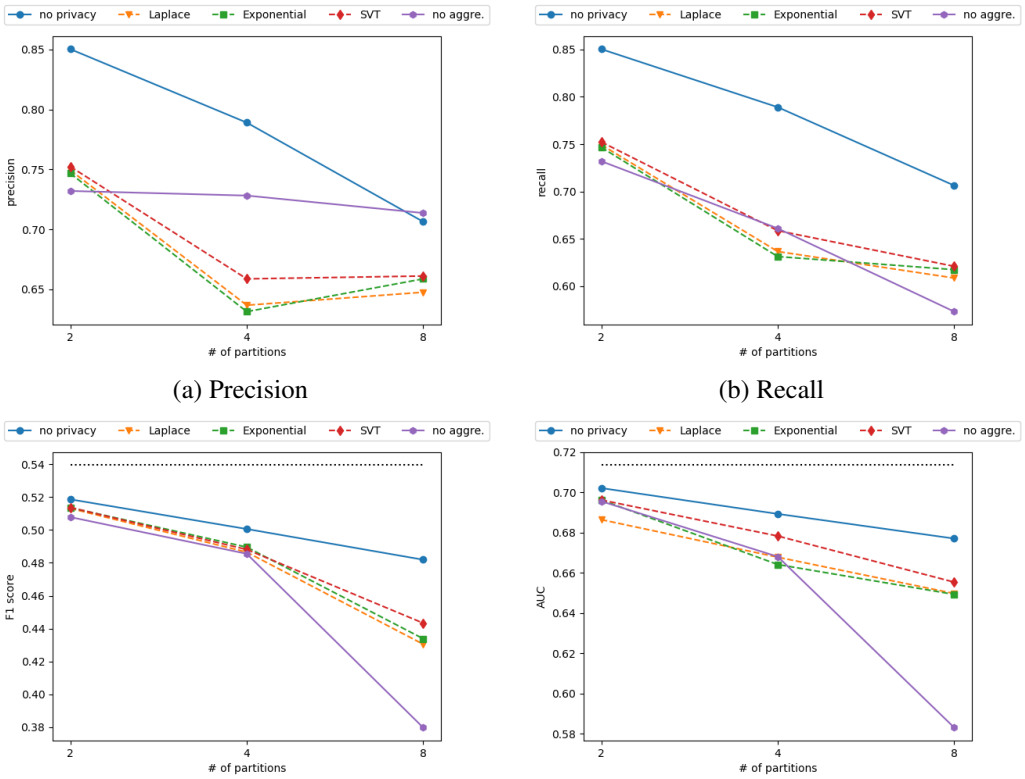


(a) F1 score (b) AUC

Figure 3: The F1 score and AUC results of risk prediction in *n=1* setting. The results are using the SPM-based framework with no aggregation with various top k values without any DP mechanisms. Note that top k means selecting top k discriminative patterns from each class, hence in total, we are using $2 * k$ patterns as the feature representation.

Table 1: The number of discriminative patterns discovered. The reported numbers are the average of partition settings out of 5 different random partitions with 3 different train-test split.

Model	2 partitions		4 partitions		8 partitions	
	CVD patterns	non-CVD patterns	CVD patterns	non-CVD patterns	CVD patterns	non-CVD patterns
No aggregation	100	100	85	100	29	100
No privacy	100	100	100	100	100	100
Laplace	100	100	98	100	82	100
Exponential	100	100	98	100	86	100
SVT	100	100	100	100	90	100



(a) Precision (b) Recall

(c) F1 score (d) AUC

Figure 4: The results show the impact of applying various DP mechanisms in various partition settings. The results are reported on the average of 5 trials of different partition settings, and 3 different train-test split. For recoverability, precision and recall are used and the results are compared with *n=1* setting (w/o DP) to check the percentage of the discriminative patterns being found from each partition. F1 score and AUC is used to evaluate the predictive power. The black dotted line in (c) and (d) are the results of *n=1* setting. All the results are using $k = 100$.

To select the best top k used throughout the remaining experiments, we evaluated k between 40 to 700. Figure 3 illustrates the results on the F1 score and AUC using various top k without any DP mechanisms on *no aggregation* framework with 1 partition. The top k in the x-axis denotes k number of discriminative patterns from each class, thus 100 means, in total, 200 patterns are used as a feature representation. From $k = 50$ to $k = 100$, both results improve rapidly while after $k = 100$, F1 score improvement becomes marginal and AUC gradually increases. This means that not all discriminative patterns are useful for risk prediction. Thus, throughout the remainder of the experiments, we fix k to be 100.

Impact of Partitions. Figure 4 summarizes the recoverability and predictive power of the resulting patterns for each

partition setting. To compute the precision and recall, we use the top 100 discriminative patterns from each partition setting (excluding *n=1*) as 'the 'retrieved" set in the Equation (2) and (3), while the top 100 discriminative patterns from *n=1* are used as "relevant".

As shown in Figure 4(a), precision stays constant for *no aggregation*. To better understand this, the number of discriminative patterns obtained by each partition setting is shown in Table 1. From the *no aggregation* row in the table, we observe that less than 100 discriminative patterns are returned when the number of partitions exceeds 2. Moreover, the number of CVD discriminative patterns decreases as the partition increases. As the "retrieved" patterns in the denominator become smaller in Equation (2), the precision should increase. However, the precision for *no aggregation* stays constant, and this indicates that the recoverability of CVD patterns is low and the precision of *no aggregation* is more related to non-CVD patterns because the number of CVD discriminative patterns decreases while the number of non-CVD discriminative patterns stays the same as the number of partition increases. On the other hand, for *no privacy*, the precision decreases as the number of partition increases, and as shown in Table 1, all the partition settings return the same number of discriminative patterns. This indicates that partitioning results in the loss of some important patterns (patterns with high support count in *n=1* setting). From Figure 4(b), we observe that the recall decreases as the number of partition increases for both *no aggregation* and *no privacy*. Similar to the precision of *no privacy*, the size of $|relevant \cap retrieved|$ is decreasing by failing to extract important patterns, thus recall decreases.

Both F1 score and AUC decrease as the number of partition increases as shown in Figures 4(c) and (d) which follow a similar trend as recall. This shows the importance of discriminative patterns and indicates that as the number of partition increases, more important discriminative patterns are being lost. This is especially true for *no aggregation*, as it is not using global discriminative patterns. The low recoverability of CVD discriminative patterns for *no aggregation* causes the predictive power to decrease. From the Figures 4(c) and (d), a larger number of partitions results in a marginal decrease in terms of predictive power for *no privacy*, and also shows that there is a marginal decrease compared to *n=1*. However, for *no aggregation*, predictive power drastically decreases as the number of partition increases, and this shows the importance of using global discriminative patterns by aggregation.

Impact of Differential Privacy. We kept the same partition settings (*i.e.*, 2, 4, 8) and evaluated the impact of various DP mechanisms. Figure 4 shows the results of applying various DP mechanisms with $\epsilon = 0.1$ and are denoted as *Laplace*, *Exponential*, and *SVT*. Figure 4(a) shows that all privacy-preserving frameworks show a similar trend but different from *no privacy*. From Table 1, we can see that the number of discriminative patterns decreases as the number of partition increases for all three DP mechanisms. And the increment of precision from *n=4* to *n=8* is a result of the number of discriminative patterns (or "retrieved" patterns) in the denominator becoming smaller in Equation (2). The decrease of precision from *n=2* to *n=4* for all DP mechanisms occurs as not all important global discriminative patterns are obtained from aggregation because the number of discriminative patterns returned is close to or equal to 100 as shown in Table 1. For recall shown in Figure 4(b), it follows a similar trend as *no privacy* which was explained previously.

The predictive performance in terms of F1 score and AUC are shown in Figure 4(c) and (d) respectively. They show that *no privacy* outperforms all other frameworks and suggests the importance of discriminative patterns. In other words, as the framework can discover more important discriminative patterns, the predictive power increases. For all DP mechanisms, they show a similar trend in predictive power which is decreasing as the number of partition increases. By comparing the results with *no privacy* and DP mechanisms, it shows that there is a trade-off by having a privacy guarantee, however, there is a less sharp loss than *no aggregation*. For the AUC of *n=4*, *no aggregation* has a higher score than *Laplace* and *Exponential*. This suggests that the framework using *Laplace* and *Exponential* mechanisms discover less important discriminative patterns which result in a lower score than *no aggregation*. In other words, by applying the *Laplace* and *Exponential* mechanisms, important discriminative patterns are discarded and discriminative patterns with low support counts are returned. And for *SVT*, although it has a similar recall with *no aggregation*, it uses 15 more CVD discriminative patterns than *no aggregation*, thus, having higher AUC. For *n=2* setting, as shown in Figure 4(d), *Exponential*, *SVT*, and *no aggregation* has similar AUC score while *Laplace* has slightly lower AUC. This again emphasizes the importance of discriminative patterns as all 4 frameworks have an equal number of discriminative patterns. The difference between all performance except *no aggregation* is marginal across the number of partitions compared to the *n=1* setting. Overall, our results suggest that our framework has minor trade-offs for preserving

privacy with predictive performance.

Discussion and Conclusions

In this paper, we propose a privacy-preserving SPM-based framework with aggregation and show the effectiveness of our method. In a large-scale analysis of EHRs, protecting patients' information is an important task, and we have shown that our privacy-preserving framework has almost similar predictive power with the framework without using DP. One limitation of the work is the usage of single real-world EHRs. As disease prevalence can vary depending on location and populations, it is important to use heterogeneous populations because single EHRs typically reflect more homogeneous populations. Another limitation is the even size of the partitions. In practice, EHRs can be stored in one or more data sources but not evenly distributed. When extracting discriminative patterns, local discriminative patterns extracted from a larger data source could dominate other local patterns, resulting in their emergence as global discriminative patterns. One possible extension is to use a weight-based aggregation to prevent one set of local discriminative patterns from dominating others. The last limitation is using only the ICD-9 codes of the patient. One possible extension is to use more information such as procedure codes or prescriptions. However, using multiple information will require more computational resources for pattern extraction. Nevertheless, our framework shows promising results, thus we leave this as future work.

In conclusion, we presented the privacy-preserving SPM-based framework with aggregation for predicting CVD risk in sequences of encounters. To demonstrate the efficiency, we compared the three frameworks (no aggregation, aggregation but no privacy, aggregation with privacy) and show the importance of discriminative patterns. Our experimental results suggest that there are minor trade-offs by applying DP mechanisms. Overall, the prediction results show the effectiveness of the framework with and without applying DP using the extracted discriminative patterns.

Acknowledgements

This work was supported by National Science Foundation awards IIS-1838200 and CNS-1952192; National Institute of Health awards 1K01LM012924-01, R01LM013323-01, and R01GM118609; and CTSA award UL1TR002378.

References

1. Bai T, Zhang S, Egleston BL, Vucetic S. Interpretable representation learning for healthcare via capturing disease progression through time. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining; 2018. p. 43–51.

2. Lee EW, Ho JC. FuzzyGap: Sequential Pattern Mining for Predicting Chronic Heart Failure in Clinical Pathways. AMIA Summits on Translational Science Proceedings. 2019;2019:222.

3. Wu J, Roy J, Stewart WF. Prediction modeling using EHR data: challenges, strategies, and a comparison of machine learning approaches. Medical care. 2010;p. S106–S113.

4. Henderson J, He H, Malin BA, Denny JC, Kho AN, Ghosh J, et al. Phenotyping through Semi-Supervised Tensor Factorization (PSST). In: AMIA Annual Symposium Proceedings. vol. 2018. American Medical Informatics Association; 2018. p. 564.

5. Warren JL, Harlan LC, Fahey A, Virnig BA, Freeman JL, Klabunde CN, et al. Utility of the SEER-Medicare data to identify chemotherapy use. Medical care. 2002;40(8):IV–55.

6. Wright AP, Wright AT, McCoy AB, Sittig DF. The use of sequential pattern mining to predict next prescribed medications. Journal of biomedical informatics. 2015;53:73–80.

7. Ghosh S, Feng M, Nguyen H, Li J. Risk prediction for acute hypotensive patients by using gap constrained sequential contrast patterns. In: AMIA annual symposium proceedings. vol. 2014. American Medical Informatics Association; 2014. p. 1748.

8. Dwork C, McSherry F, Nissim K, Smith A. Calibrating noise to sensitivity in private data analysis. In: Theory of cryptography conference. Springer; 2006. p. 265–284.

9. Centers for Disease Control and Prevention. National diabetes statistics report, 2020. Atlanta, GA: Centers for Disease Control and Prevention, U.S. Dept of Health and Human Services; 2020.

10. Dieren Sv, Beulens JWJ, Schouw YTvd, Grobbee DE, Neal B. The global burden of diabetes and its complications: an emerging pandemic. European Journal of Cardiovascular Prevention & Rehabilitation. 2010 May;17(1_suppl):s3–s8.

11. Heidenreich PA, Trogdon JG, Khavjou OA, Butler J, Dracup K, Ezekowitz MD, et al. Forecasting the future of cardiovascular disease in the United States: a policy statement from the American Heart Association. Circulation. 2011 Mar;123(8):933–944.

12. Association AH. Cardiovascular disease & diabetes; 2017. http://www.heart.org/HEARTORG/Conditions/More/Diabetes/WhyDiabetesMatters/Cardiovascular-Disease-Diabetes_UCM_313865_Article.jsp#.WckMddOGMUE.

13. Feldman DI, Valero-Elizondo J, Salami JA, Rana JS, Ogunmoroti O, Osondu CU, et al. Favorable cardiovascular risk factor profile is associated with lower healthcare expenditure and resource utilization among adults with diabetes mellitus free of established cardiovascular disease: 2012 Medical Expenditure Panel Survey (MEPS). Atherosclerosis. 2017 Mar;258:79–83.

14. Fournier-Viger P, Lin JCW, Kiran RU, Koh YS, Thomas R. A survey of sequential pattern mining. Data Science and Pattern Recognition. 2017;1(1):54–77.

15. McSherry F, Talwar K. Mechanism design via differential privacy. In: 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07). IEEE; 2007. p. 94–103.

16. Lyu M, Su D, Li N. Understanding the sparse vector technique for differential privacy. arXiv preprint arXiv:160301699. 2016;.

17. Yang Q, Liu Y, Chen T, Tong Y. Federated machine learning: Concept and applications. ACM Transactions on Intelligent Systems and Technology (TIST). 2019;10(2):1–19.

18. Geyer RC, Klein T, Nabi M. Differentially private federated learning: A client level perspective. arXiv preprint arXiv:171207557. 2017;.

19. Truex S, Baracaldo N, Anwar A, Steinke T, Ludwig H, Zhang R, et al. A hybrid approach to privacy-preserving federated learning. In: Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security; 2019. p. 1–11.

20. Choudhury O, Gkoulalas-Divanis A, Salonidis T, Sylla I, Park Y, Hsu G, et al. Differential privacy-enabled federated learning for sensitive health data. arXiv preprint arXiv:191002578. 2019;.

21. Geraci JM, Ashton CM, Kuykendall DH, Johnson ML, Wu L. International Classification of Diseases, 9th Revision, Clinical Modification codes in discharge abstracts are poor measures of complication occurrence in medical inpatients. Medical care. 1997;p. 589–602.

22. Fournier-Viger P, Gomariz A, Campos M, Thomas R. Fast vertical mining of sequential patterns using co-occurrence information. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer; 2014. p. 40–52.

23. Salvemini E, Fumarola F, Malerba D, Han J. Fast sequence mining based on sparse id-lists. In: International Symposium on Methodologies for Intelligent Systems. Springer; 2011. p. 316–325.

24. Fumarola F, Lanotte PF, Ceci M, Malerba D. CloFAST: closed sequential pattern mining using sparse and vertical id-lists. Knowledge and Information Systems. 2016;48(2):429–463.

25. Fournier-Viger P, Lin JCW, Gomariz A, Gueniche T, Soltani A, Deng Z, et al. The SPMF open-source data mining library version 2. In: Joint European conference on machine learning and knowledge discovery in databases. Springer; 2016. p. 36–40.