Temporal Phenotyping from Longitudinal Electronic Health Records: A Graph Based Framework

Chuanren Liu[1], Fei Wang[2], Jianying Hu[3], Hui Xiong[4]

[1]Decision Sciences & MIS Department, Drexel University
[2]Department of Computer Science and Engineering, University of Connecticut
[3]Healthcare Analytics Research Group. IBM T. J. Watson Research Center
[4]Management Science and Information Systems Department, Rutgers University
chuanren@xminer.org, fei_wang@uconn.edu, jyhu@us.ibm.com, hxiong@rutgers.edu

ABSTRACT

The rapid growth in the development of healthcare information systems has led to an increased interest in utilizing the patient Electronic Health Records (EHR) for assisting disease diagnosis and phenotyping. The patient EHRs are generally longitudinal and naturally represented as medical event sequences, where the events include clinical notes, problems, medications, vital signs, laboratory reports, etc. The longitudinal and heterogeneous properties make EHR analysis an inherently difficult challenge. To address this challenge, in this paper, we develop a novel representation, namely the temporal graph, for such event sequences. The temporal graph is informative for a variety of challenging analytic tasks, such as predictive modeling, since it can capture temporal relationships of the medical events in each event sequence. By summarizing the longitudinal data, the temporal graphs are also robust and resistant to noisy and irregular observations. Based on the temporal graph representation, we further develop an approach for temporal phenotyping to identify the most significant and interpretable graph basis as phenotypes. This helps us better understand the disease evolving patterns. Moreover, by expressing the temporal graphs with the phenotypes, the expressing coefficients can be used for applications such as personalized medicine, disease diagnosis, and patient segmentation. Our temporal phenotyping framework is also flexible to incorporate semi-supervised/supervised information. Finally, we validate our framework on two real-world tasks. One is predicting the onset risk of heart failure. Another is predicting the risk of heart failure related hospitalization for patients with COPD pre-condition. Our results show that the diagnosis performance in both tasks can be improved significantly by the proposed approaches. Also, we illustrate some interesting phenotypes derived from the data.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data Mining*

KDD '15, August 10-13, 2015, Sydney, NSW, Australia.
© 2015 ACM. ISBN 978-1-4503-3664-2/15/08…$15.00.
DOI: http://dx.doi.org/10.1145/2783258.2783352

General Terms

Algorithms, Application

Keywords

Temporal Graph; Temporal Phenotyping; Regularization; Electronic Health Records

1. INTRODUCTION

With the rapid development of computer software and hardware technologies, various kinds of healthcare data, including patient Electronic Health Records (EHR), research and development data in pharmaceutical companies, vital information captured from wearable devices, and even social media data from online health forums and websites are becoming more and more available. Effective mining of those data to get actionable insights is now a hot research topic and the "data driven healthcare" is believed to be an emerging trend for improving the quality of care delivery [1, 18].

Patient EHRs [8], defined as systematic collection of patient health information in electronic form, is one of the major carriers for conducting data driven healthcare research. However, there are various challenges if we work directly with EHRs, such as sparsity, noisiness, heterogeneity, bias, etc [7]. To address these challenges, before going into the stage of detailed applications, we should first do *electronic phenotyping*, which is basically a feature extraction process transforming the raw EHR data into clinically relevant features [7, 22]. There has been quite a few existing electronic phenotyping works. For example, Ho *et al.* [6] proposed a tensor factorization based approach for high throughput phenotyping. Lasko *et al.* [13] proposed a deep learning method for obtaining phenotypes from lab value signals. Kale *et al.* [9] applied deep learning to discover the physiomes from the physiological streams obtained in Pediatric Intensive Care Unit (PICU). Zhou *et al.* [29] proposed an optimization based technology for discovering the phenotypes within which the raw medical features have similar evolving patterns. For all these works, the authors either define a phenotype as some evolving pattern on the values of a specific medical feature (e.g., lab test or physiological stream), or a group of medical features (e.g., diagnosis, medication or both). Another important type of phenotype is the temporal pattern across different medical features. The existing approaches on temporal phenotype identification are mostly based on sequential pattern mining [4, 23] or temporal abstraction [21, 24, 25]. One major challenge for these methodologies working on EHRs is *phenotype explosion*, which is the phenomenon that too many phenotypes are identified from the patient EMR corpus with improper support threshold. One could try to solve this problem by increasing the support threshold value used by sequential pattern mining but the mined phenotypes are then typically trivial. Therefore there is an urgent need on an effective way to identify a reasonable number of clinical meaningful phenotypes.

One cause of pattern explosion for traditional approaches is the sequence based representation: the patient EHRs are so complicated and the high variability within which generates a huge number of sequential patterns when the support threshold is relatively lower. In this paper, we propose a novel graph based representation for patient EHRs, where the EHRs of every patient is represented as a graph. The nodes in the graph are the medical events (i.e., diagnosis, medications, lab tests, etc.). The edges encode the temporal relationships among the events in the EHRs of the corresponding patient. Every edge points from an event to another event that took place later in time. A weight will also be associated with each edge, which reflects the average duration between the two events in EHRs. A basis learning framework is then developed to identify the temporal phenotypes that can be used to compose all those temporal graphs. We present several concrete instantiations of such framework and validate its effectiveness on a real-world EHR data warehouse both quantitatively and qualitatively.

It is worthwhile to highlight the following aspects of the proposed graph based framework:

- The framework represents patient EHRs as temporal graphs. Comparing to sequence based representation, such temporal graph is more compact, which makes the downstream phenotyping procedure more efficient.
- With graph representation, the detected phenotypes are subgraphs instead of subsequences. Each subgraph is a natural aggregation of a set of subsequences. This effectively alleviates the pattern explosion problem while at the same time retains the interpretability of the mined phenotypes.
- The framework is flexible. We provide concrete instantiation examples of such framework in a complete unsupervised scenario, as well as in scenarios where we incorporate expertise knowledge as semi-supervised and supervised regularizers.
- The framework is validated on a real-world EHR data warehouse under two clinical scenarios. One is early detection of Congestive Heart Failure (CHF). The other is hospital readmission prediction of CHF patients with Chronic Obstructive Pulmonary Disease (COPD) preconditions. Both of them are important clinical problems that have been widely studied in medical research.

2. RELATED WORK

This section reviews the existing work on electronic phenotyping and temporal knowledge representation, which are closely related to the research proposed in this paper.

2.1 Electronic Phenotyping

Genotype and *phenotype* are two basic concepts in biology and medicine. Genotype is the genetic makeup of a cell, an organism, or an individual usually with reference to a specific characteristic under consideration, which encodes an organism's full hereditary information. Phenotype is an organism's actual observed properties, such as morphology, development, or behavior. The systematic description of phenotype variation has gained increasing importance since the discovery of the causal relationship between a genotype placed in a certain environment and a phenotype. Accurate phenotyping has the potential to be the bridge between studies that aim to advance the science of medicine (such as a better understanding of the genomic basis of diseases), and studies that aim to advance the practice of medicine (such as phase IV surveillance of approved drugs) [26].

Electronic phenotyping refers to the process of identifying phenotypes from patient EHRs, which, in the word of data mining, is the procedure of extracting clinically relevant features. There are some existing electronic phenotyping works. For example, Ho *et al.* [5, 6] formulates the patient EHRs as tensors, wherein every mode represents a specific type of medical event. The entries in the tensor encode the interaction of those features (e.g., the frequency of a medication and a primary diagnosis). Then they proposed a tensor factorization based approach for identification of the phenotypes. Zhou *et al.* [29] formulates EHRs as temporal matrices with medical events as one dimension and time as the other dimension. They propose an optimization based technology for discovering the phenotypes within which the raw medical features have similar time-evolving patterns. Lasko *et al.* [13] proposed a deep learning method for obtaining phenotypes from continuous lab value signals, where they first adopted Gaussian process regression to impute the missing lab test values. Kale *et al.* [9] applied deep learning to discover the physiomes from the physiological streams obtained in Pediatric Intensive Care Unit (PICU). For all these works, they either define a phenotype as some evolving pattern on the values of a specific medical feature (e.g., lab test or physiological stream), or a group of medical features (e.g., diagnosis, medication or both). They did not consider the temporal relationships across different medical events, which could be crucial as it suggests important information on the impending disease conditions.

2.2 Temporal Knowledge Representation

Knowledge representation from temporal data is a hot research topic in data mining. A lot of research has been done along this line. In general temporal data can be categorized as either continuous or discrete. For knowledge representation of continuous time data, one of the most popular approaches is to transform the multivariate continuous time series into discrete symbolic representations (string, nominal, categorical, and item sets). Popular approaches include Piecewise Linear Approximation (PLA) [12], Adaptive Piecewise Constant Approximation (APCA) [10], Symbolic Aggregate approXimation (SAX) [16], Piecewise Aggregate Approximation (PAA) [11], etc. One can refer to [16] for a survey on these approaches.

For knowledge representation of discrete time series data, Mörchen *et al.* [19, 20] proposed the Time Series Knowledge Representation (TSKR) as a pattern language (grammar) for temporal knowledge discovery from multivariate time series and symbolic interval data, where the temporal knowledge representation is in the form of symbolic languages and grammars that have been formulated as a means to perform intelligent reasoning and inference from time-dependent event sequences. More recently, Wang *et al.* [28]

proposed a convolutional framework to extract temporal signatures in discrete time data using the Temporal Event Matrix Representation (TEMR), which is shown to have wide applicability to a variety of data and application domains that involve large-scale longitudinal data.

The temporal graph we propose in this paper provides an alternative way to represent the temporal knowledges contained in discrete time data. The temporal graphs capture temporal structures hidden in the sequences in a more compact way, where the nodes in the graph are events in the EHR and the directed edges encode the temporal relationships between pairwise events. In the temporal graph, the events missing in patient EHRs will not appear, and the repeated pairwise events with the same ordering will only appear once. With this representation, the temporal graph is robust and resistant to sparse, noisy, and irregular observations. Moreover, this representation is very intuitive and highly interpretable, because one can easily understand the temporal relationships among different medical events in patient EHRs. Another advantage is that with graph based representation, the detected phenotypes (or patterns) will also be in the form of graphs, which can be viewed as a nature aggregation of sequential patterns. In this way, we can effectively alleviate the pattern explosion problem.

3. METHODOLOGY

In this section we will introduce the details of our temporal graph based framework for phenotype identification from patient EHRs. First we present the basic definition of temporal graph and how it is constructed.

3.1 Temporal graph construction

Suppose we have a set of event sequences $\{s_n : n = 1, \cdots, N\}$ where N is the number of sequences. Each event sequence is denoted by $s_n = ((x_{nl}, t_{nl}) : l = 1, \cdots, L_n)$ where L_n is the length of s_n. In other words, we can observe event x_{nl} at time t_{nl} in the sequence s_n. We let the events $x_{nl} \in \{1, \cdots, M\}$ and $t_{np} \leq t_{nq}$, for all $p < q$. We have one example of the medical event sequences of potential patients in Figure 1. With the observed event sequences, inspired by Liu et al. [17], we construct the following temporal graph for each sequence s_n:

Definition 1 (Temporal graph). *The temporal graph G^n of sequence s_n is a directed and weighted graph with our event set as its node set $\{1, \cdots, M\}$, where the weight of the edge from node i to node j is defined as*

$$W_{ij}^n = \frac{1}{L_n} \sum_{1 \leq p \leq q \leq L_n} [x_{np} = i \wedge x_{nq} = j] \kappa (t_{nq} - t_{np}), \quad (1)$$

where $\kappa(\cdot)$ is a non-increasing function.

Note that $\kappa(\cdot)$ is a non-increasing function, thus the more often events i and j appear close to each other in s_n, the higher the W_{ij}^n is. In this paper, we use the exceedance of the Exponential distribution to construct the temporal graph:

$$\kappa(\delta) = \begin{cases} \exp(-\delta/r) & \delta \leq \Delta \\ 0 & \delta > \Delta \end{cases} \quad (2)$$

In other words, we compute a smaller edge weight for a larger time interval δ, when $\delta \leq \Delta$. Otherwise, we ignore the event pairs when the events happened with a time interval δ larger than the threshold Δ.

In Figure 2, we present the temporal graph of the event sequence in Figure 1. In the sequence, we have 6 observations of 4 unique events. We show the interval between event-happening timestamps along the ordered edge between the observations. In the graph, we use the edge width to signify the weighted computed with the input sequence. As can be seen, the parameters Δ, r control the locality of the edge computation in the temporal graph. Namely, a larger r captures the similarities among events in a longer temporal range, and potentially increases the connectivity of the temporal graph. A small r only considers closely adjacent symbols as similar, and makes the temporal graph more spread. In the extreme case when r approaches infinity, W^n becomes an almost constant matrix, since all appearing event pairs will be fully and equally connected.

The parameters (Δ and r) can be selected according to the application. For example, if there is little correlation between events happened with a time interval larger than 3 months, then we can let $\Delta = 3$ months. The scaling parameter r can also be empirically set to be the average time interval between consecutive events. In the example, we use $\Delta = 3$ months and $r = 5$ days. In our empirical study on real-world EHR data warehouse, we optimize r based on the phenotyping performance in specific applications.

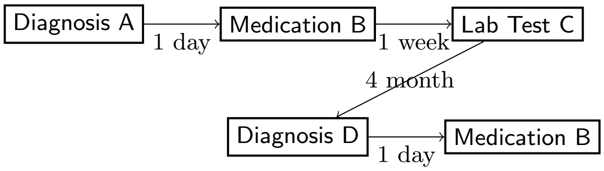


Figure 1: One example of medical event sequence of one subject (potential patient).

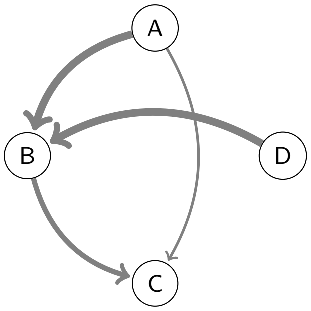


Figure 2: The temporal graph of event sequence in Figure 1.

3.2 Temporal phenotyping

With all the constructed temporal graphs, we want to identify the temporal phenotypes that can be used to best explain the observations. Our idea is to compute the graph basis as the temporal phenotypes which can be used to reconstruct the observed temporal graphs. In Figure 3, we have one simplified example, where we have three graph basis, and one observed graph can be expressed as the average of the first two basis. In practice, we do not know the basis in the beginning, and our temporal phenotyping is exactly the process identifying the unknown graph basis with the observed temporal graphs.

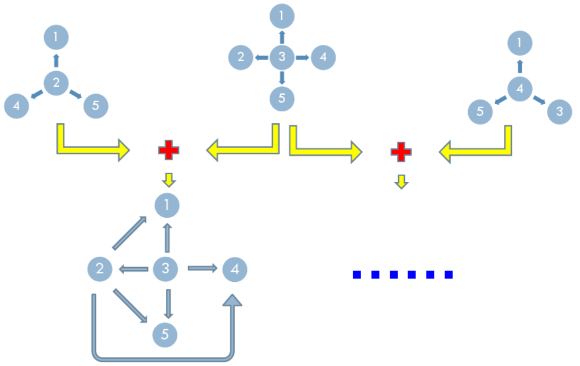


Figure 3: Example of temporal phenotyping.

We call the resultant graph basis as temporal phenotypes, since they are derived from the temporal graphs, and the temporal phenotypes capture evolving patterns of the health conditions hidden in the event sequences. To be specific, suppose we have constructed the temporal graph G^n for each sequence s_n, and G^n is associated with the adjacency weight matrix $W^n \in \mathbb{R}^{M \times M}$. To reconstruct G^n, we assume there are K graph basis $B^k \in \mathbb{R}^{M \times M}$ for $k = 1, 2, \cdots, K$, which can be used to approximate the adjacency matrix W^n:

$$W^n = \sum_{k=1}^{K} A_{nk} B^k,$$

where $A \in \mathbb{R}^{N \times K}$ is the matrix of reconstruction coefficients. To compute the optimal graph basis and the reconstruction coefficients, we minimize the total reconstruction error:

$$\mathcal{J}(A, B) = \frac{1}{2} \sum_{n=1}^{N} \|W^n - \sum_{k=1}^{K} A_{nk} B^k\|_F^2, \qquad (3)$$

where $\| \cdot \|_F$ is the matrix Frobenius norm.

To make the solutions more interpretable, we also consider two constraints on the reconstruction coefficients in A and the graph basis B^k for $k = 1, 2, \cdots, K$. The first constraint is about the non-negativity, i.e., $B^k \geq 0$ for all k, since our original temporal graphs are non-negative. The second constraint requires $A \geq 0$ and $\sum_k A_{nk} = 1$, for $n = 1, \cdots, N$, which make the rows of A to be valid multinomial distribution. In this way, we can quantify each patient by the temporal phenotypes with probabilities which can be in turn used for personalized medicine, patient segmentation, and disease diagnosis.

3.3 Regularization

As we introduced earlier, the reconstruction coefficients in A can be used for a various of applications. In particular, for the medical diagnosis application, our goal is to derive informative features to improve the diagnosis performance, i.e., the classification of control/case groups for the patients. To this end, we extend the temporal phenotyping for temporal graphs with regularization $\Omega(A) \geq 0$:

$$\mathcal{J}(A, B) = \frac{1}{2} \sum_{n=1}^{N} \|W^n - \sum_{k=1}^{K} A_{nk} B^k\|_F^2 + \lambda \Omega(A), \qquad (4)$$

where $\lambda \geq 0$ is the parameter controlling the degree of regularization. In the following, we propose several regularizations as $\Omega(A)$ to incorporate additional knowledge on the patients under study.

3.3.1 *Similarity based regularization*

In the first case, we have limited supervision [2] such as implicit similarity links between patients who are from the same group (case or control). We can encourage the linked patients to have similar phenotyping representations in A using the following regularization:

$$\Omega(A) = \frac{1}{2} \sum_{n_1, n_2} \frac{1}{2} \|A_{n_1} - A_{n_2}\|^2 S_{n_1 n_2},$$

where $S \in \mathbb{R}^{N \times N} > 0$ is symmetric matrix encoding the similarity information. Note that, when S is asymmetric, we can just equivalently replace S with $(S + S')/2$ without changing $\Omega(A)$. It follows that $S_{n_1 n_2} = S_{n_2 n_1}$ and

$$\Omega(A) = \frac{1}{2} \operatorname{tr}(A' L A), \qquad (5)$$

where $L = D - S$ and D is the diagonal degree matrix such that $D_{nn} = \sum_{n'} S_{nn'}$. Note that, some rows/columns of S may be completely zero if we do not have knowledge about the corresponding patients, e.g., the instances in the test set.

3.3.2 *Model based regularization*

In the second case, we have access to the group information of the patients. We let $Y_n = 1$ if the n-th patient is from the case group and $Y_n = -1$ if the patient is from the control group. With the explicit label information, we can define the regularization $\Omega(A)$ directly with a discriminative model $\Pr(A_n, Y_n | \mathcal{H})$:

$$\Omega(A) = -\frac{1}{|\mathcal{L}|} \sum_{n \in \mathcal{L}} \log \Pr(A_n, Y_n | \mathcal{H}), \qquad (6)$$

which is termed as average log-loss in the literature. Here, $\mathcal{L}$ is the training set where we have label Y_n for $n \in \mathcal{L}$.

One particular choice for the discriminative model we can use for the case/control classification of patients is the logistic regression:

$$\Pr(A_n, Y_n | \mathcal{H}) = \frac{1}{1 + \exp(-Y_n f(A_n))},$$

where the linear model:

$$\mathcal{H} : A_n \mapsto f(A_n) = A_n \Theta + \theta$$

and (Θ, θ) are parameters in the model $\mathcal{H}$. It follows that

$$\log \Pr(A_n, Y_n | \mathcal{H}) = -\log(1 + \exp(-Y_n f(A_n))).$$

In addition to the log-loss for probabilistic model, other loss terms can also be used with the linear model $\mathcal{H}$. We consider the hinge loss for (A_n, Y_n):

$$\operatorname{loss}(A_n, Y_n | \mathcal{H}) = \max\{0, 1 - Y_n f(A_n)\},$$

and the general model based regularization:

$$\Omega(A) = \frac{1}{|\mathcal{L}|} \sum_{n \in \mathcal{L}} \operatorname{loss}(A_n, Y_n | \mathcal{H}). \qquad (7)$$

3.4 Implementation

We give the implementation of the regularized temporal phenotyping problems, since the un-regularized problems are special cases with $\lambda = 0$.

3.4.1 Similarity based regularization

We iteratively solve A and B for the temporal phenotyping (Equation 4) with the similarity based regularization (Equation 5). When updating A with B fixed, we adopt the projected gradient descent approach. Specifically, we have the gradient of A:

$$\frac{\partial \mathcal{J}}{\partial A} = A\langle B \otimes B\rangle - \langle W \otimes B\rangle + \lambda \cdot LA,$$

where we define $\langle B \otimes B\rangle \in \mathbb{R}^{K\times K}$ such that

$$\langle B \otimes B\rangle_{k_1 k_2} = \sum_{i=1}^{M}\sum_{j=1}^{M} B_{ij}^{k_1} B_{ij}^{k_2},$$

and similarly, $\langle W \otimes B\rangle_{k_1 k_2} = \sum_{i=1}^{M}\sum_{j=1}^{M} W_{ij}^{k_1} B_{ij}^{k_2}$. With the gradient of the current solution A, we update

$$A \leftarrow \operatorname{proj}_{\mathrm{splx}}(A - \alpha\frac{\partial \mathcal{J}}{\partial A}),$$

where $\operatorname{proj}_{\mathrm{splx}}(A)$ projects each row A_n of A to the simplex such that $A_{nk} \geq 0$ and $\sum_k A_{nk} = 1$. The step size α is selected using the Armijo rule, such that

$$\frac{1}{2}\operatorname{tr}(\Delta\langle B \otimes B\rangle\Delta') + \frac{\lambda}{2}\operatorname{tr}(\Delta' L\Delta) + (1-\sigma)\operatorname{tr}((\frac{\partial \mathcal{J}}{\partial A})'\Delta) \leq 0,$$

where $\Delta = \operatorname{proj}_{\mathrm{splx}}(A - \alpha\frac{\partial \mathcal{J}}{\partial A}) - A$. A common choice of σ is 0.01 [15].

With A fixed, we can update the basis B^k for all k simultaneously, but in an element-wise manner. To be specific, for any event-pair (i, j), $1 \leq i, j \leq M$, we can update all B_{ij}^k, $1 \leq k \leq K$ at the same time, with the following subproblem:

$$\min_{B_{ij}^*} \frac{1}{2}\sum_{n=1}^{N}(W_{ij}^n - \sum_{k=1}^{K} A_{nk}B_{ij}^k)^2.$$

This subproblem is exactly non-negative regression fitting the data A_{n*} with dependent variable W_{ij}^n for $1 \leq n \leq N$. Note that, the updating process of different event-pairs (i, j) in graph basis are independent to each other during this optimization step, and the M^2 subproblem can be solved in parallel.

3.4.2 Model based regularization

We have three sets of parameters A, B and (Θ, θ) in the temporal phenotyping (Equation 4) with the model based regularization (Equation 6). First of all, the updating of B with A and (Θ, θ) fixed is the same non-negative least square regression problem presented in Section 3.4.1. Second, when updating A with B and (Θ, θ) fixed, we note that the rows of A are independent to each other during this optimization step, and thus we optimize each of them separately or in parallel. For the n-th row A_n, if $n \notin \mathcal{L}$, the problem reduces to constrained least square regression to minimize $\|W^n - \sum_{k=1}^{K} A_{nk}B^k\|_F^2$ subject to $A_n \geq 0$ and $\sum_k A_{nk} = 1$. This can be solved using the same projected gradient descend approach presented in Section 3.4.1 (with $L = 0$).

For the updating of A_n if $n \in \mathcal{L}$, we have the gradient:

$$\frac{\partial \mathcal{J}}{\partial A_n} = A_n\langle B \otimes B\rangle - \langle W^n \otimes B\rangle - \frac{\lambda}{|\mathcal{L}|}Y_n\frac{1}{1+\exp(Y_n f(A_n))}\Theta'.$$

In this case, the general Armijo rule for selecting step size α to update $A_n \leftarrow \operatorname{proj}_{\mathrm{splx}}(A_n - \alpha\frac{\partial \mathcal{J}}{\partial A_n})$ is:

$$\mathcal{J}(A_n + \Delta_n) - \mathcal{J}(A_n) \leq \sigma\langle\frac{\partial \mathcal{J}}{\partial A_n}, \Delta_n\rangle,$$

where $\Delta_n = \operatorname{proj}_{\mathrm{splx}}(A_n - \alpha\frac{\partial \mathcal{J}}{\partial A_n}) - A_n$.

For the updating of (Θ, θ) in the log-loss regularization, we optimize:

$$\min_{\Theta,\theta} -\frac{1}{|\mathcal{L}|}\sum_{n\in\mathcal{L}}\log\Pr(A_n, Y_n|\mathcal{H}) + \frac{1}{2C}\|\mathcal{H}\|^2,$$

where $\|\mathcal{H}\|^2 = \|\Theta\|^2 + \theta^2$. This is just the logistic regression modelling with training data $\{(A_n, Y_n)|n \in \mathcal{L}\}$.

To implement the hinge loss regularization in Equation 7, in comparison with Equation 6, the only differences during the whole updating process happen when updating A_n for $n \in \mathcal{L}$ and updating (Θ, θ). To update A_n for $n \in \mathcal{L}$, since the hinge loss is non-differentiable, we take the general proximal gradient optimization approach. Specifically, we optimize:

$$\min_{A_n\in\mathbb{R}^{1\times K}} \frac{1}{2}\|W^n - \sum_{k=1}^{K} A_{nk}B^k\|_F^2 + \frac{\lambda}{|\mathcal{L}|}\operatorname{loss}(A_n, Y_n|\mathcal{H}) + \operatorname{splx}(A_n),$$

where $\operatorname{splx}(A_n) = 0$ if $\sum_k A_{nk} = 1$ and $A_{nk} \geq 0$, $\operatorname{splx}(A_n) = \infty$ otherwise. This formulation involves a sum of convex differentiable term and convex non-differentiable regularizations, which make the problem non trivial. Fortunately, the proximal gradient optimization approach can help if we can easily compute the proximal operators for each of the regularizations separately. The general proximal operator prox_R for a convex regularization R at a point P is defined as:

$$\operatorname{prox}_R(P) = \arg\min_Q \frac{1}{2}\|P - Q\|^2 + R(Q).$$

It follows that

$$\operatorname{prox}_{\tau\,\mathrm{loss}}(A_n) = A_n + Y_n\operatorname{proj}_{[0,\tau]}(\frac{1 - Y_n f(A_n)}{\|\Theta\|^2})\Theta',$$

$$\operatorname{prox}_{\mathrm{splx}}(A_n) = \operatorname{proj}_{\mathrm{splx}}(A_n).$$

where $\operatorname{proj}_{[0,\tau]}(\alpha) = \begin{cases} \alpha & \alpha \in [0, \tau] \\ 0 & \alpha < 0 \\ \tau & \alpha > \tau \end{cases}$. With the proximal operators, we present the algorithm updating A_n in Algorithm 1. For the step size α in the algorithm, we apply the line search with the Armijo rule:

$$\frac{1}{2}\langle\Delta_n\langle B\otimes B\rangle, \Delta_n\rangle + (1-\sigma)\langle A_n\langle B\otimes B\rangle - \langle W^n\otimes B\rangle, \Delta_n\rangle \leq 0,$$

where $\Delta_n = A_n^{end} - A_n^{start}$ is the difference between the end value and the start value of A_n during each repeat.

To update (Θ, θ) in the hinge loss regularization, the objective is to optimize:

$$\min_{\Theta,\theta} \frac{1}{|\mathcal{L}|}\sum_{n\in\mathcal{L}}\max\{0, 1 - Y_n(A_n\Theta + \theta)\} + \frac{1}{2C}\|\mathcal{H}\|^2,$$

where $\|\mathcal{H}\|^2 = \|\Theta\|^2 + \theta^2$. This is exactly the SVM learning with training data $\{(A_n, Y_n)|n \in \mathcal{L}\}$.

Algorithm 1 The algorithm updating A_n for $n \in \mathcal{L}$ with hinge loss regularization

```
Initialize $A_n$
repeat
    $A_n \leftarrow A_n - \alpha(A_n\langle B \otimes B\rangle - \langle W^n \otimes B\rangle)$
    $A_n \leftarrow A_n + Y_n \, \text{proj}_{[0,\alpha\frac{\lambda}{|\mathcal{L}|}]}(\frac{1-Y_n f(A_n)}{\|\Theta\|^2})\Theta'$
    $A_n \leftarrow \text{proj}_{\text{splx}}(A_n)$
until Convergence
```

4. EMPIRICAL EVALUATION

In this section, we evaluate the effectiveness of our temporal phenotyping approach on both synthetic data and real-world patient Electronic Health Record (EHR) data warehouse. We first introduce the results on synthetic data.

4.1 Synthetic Data

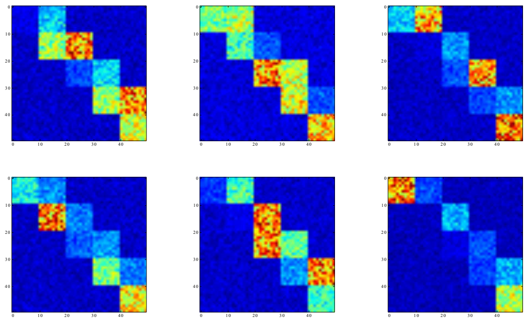


Figure 4: Examples of the synthetic data. Every sample is a 50x50 matrix generated by convex combination of the basis in Figure 5. The combination coefficients are first generated from uniform distribution within [0,1] and then normalized.

In this part, we randomly generate a set of nonnegative square matrices as the weighted adjacency matrices of the temporal graphs, and the main goal is to test the whether the phenotype identification process introduced in Section 3 can find the correct basis. The graph size are all 50x50. Figure 4 illustrates some sample adjacency matrices. Every graph is constructed by a random convex combination of the seven basis shown in Figure 5. Each basis is composed by a 50x50 background noise matrix with entries uniformly distributed in [0, 0.1], plus several 10x10 foreground matrices with entries uniformly distributed in [0, 1]. The combination coefficients for every sample are also randomly generated from [0, 1] and then normalized. We generated 1,000 samples in total.

We first validate our approach in a qualitative way, i.e., whether the method proposed in Section 3.2 can identify the correct basis, and we set the number of basis to 7. Figure 6 demonstrates the basis obtained by our algorithm. By comparing them with the true basis in Figure 5, we can see that they exactly match on the foreground patterns. There are only slight differences on the background noise.

We also tested the effectiveness of our approach quantitatively, where a binary label is associated for each of the 1,000 samples in the following way. We generate a seven dimensional decision vector, and every sample is also represented by the seven dimensional combination coefficient vector. The inner product between the decision vector and the coefficient vector will be the decision score for every sample, and the mean score over all 1,000 samples is used as the threshold. For any sample with score larger than the threshold will be assigned with label +1, otherwise they will be assigned with label -1.

We apply the different temporal phenotyping methods in Section 3 to learn the graph basis as well as the combination coefficients, and the coefficient vector will be used as the learned representation of every sample. We then randomly partition them into training and testing sets and use Support Vector Machine (SVM) to perform the prediction task (Note that, for supervised temporal phenotyping, the classification model $\mathcal{H}$ is already learned). This problem could be very difficult because both the ground truth coefficient vectors and decision vector are randomly generated. We use three measures to evaluate the prediction performance:

- **AUC**: area under the classification Receiver Operating Characteristic (ROC) curve.
- **AUPR**: area under the classification Precision-Recall (PR) curve.
- **ACC**: accuracy, i.e., ratio of true predictions.

Table 1 reported average and standard deviation of the classification performance of different approaches over 10-fold cross validation, where "Graph" is the baseline method that directly stretches the graph adjacency matrices as vectors for classification. The four methods under "Temporal Phenotyping" are the different versions of our proposed algorithms in Section 3, where "Un" is the unsupervised method in Section 3.2, "Sim" is the method with similarity based regularization in Section 3.3.1, "Logit" and "Hinge" are the approaches with logistic and hinge loss regularizations in Section 3.3.2. From the table we can observe that: (1) basis based representation can achieve better classification performance than plain graph based representation; (2) supervised phenotyping generally produces better results.

4.2 Real-World EHR Data Warehouse

In this part, we studied the effectiveness of our proposed approaches on a real-world EHR data warehouse including the records of 319,650 patients over 4 years. We use the diagnosis information of the first three digits of ICD-9 and the medication information in terms of drug ingredients to construct the EHR sequences. The temporal graphs are constructed from those sequences according to Definition 1. We will study the following two specific scenarios:

One-Year Hospitalization Prediction We identify a set of 430 Congestive Heart Failure (CHF) patients with Chronic Obstructive Pulmonary Disease (COPD) pre-condition. Among them 100 are hospitalized within one year after CHF confirmation, the rest 330 patients are not. The goal is to make use of the records 360 days prior to the CHF confirmation date to predict whether the patients will be hospitalized or not within one year after CHF confirmation. The graphical illustration of this setting is in Figure 7.

Early Prediction of CHF We first identify a set of 1127 case patients who are confirmed with Congestive Heart Failure (CHF), and then construct a set of 3850 group

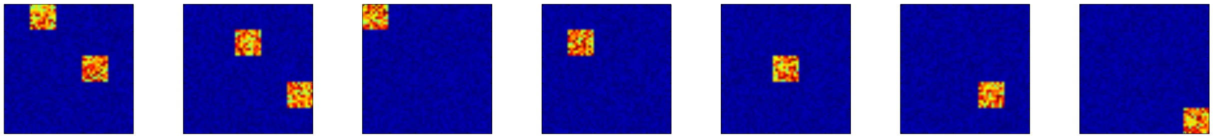


Figure 5: The basis for generating the synthetic data. Every base is a 50x50 matrix. The background noise are with values randomly generated from uniform distribution in [0,0.1]. The foreground blocks are 10x10 with values generated from uniform distribution in [0,1].

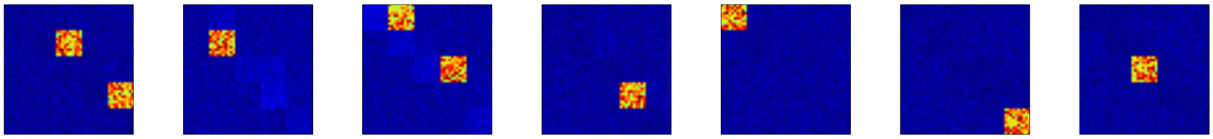


Figure 6: The basis learned by our algorithm without any regularizations. By comparing them with the basis in Figure 5 we can see they exactly match on the foreground patterns. There are only slight differences on the background noise.

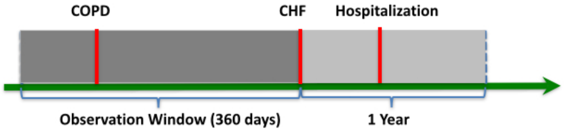


Figure 7: Experimental setting of hospitalization prediction.

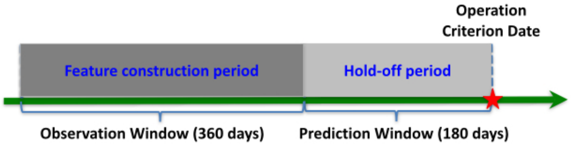
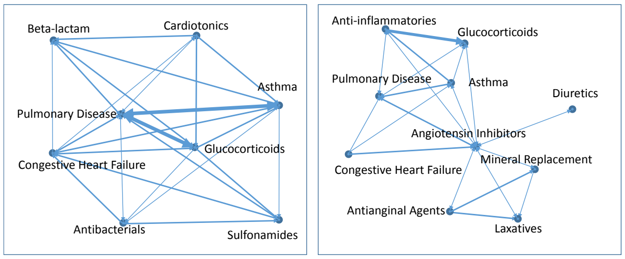

Operation
Criterion Date
Feature construction period
Hold-off period
Observation Window (360 days)
Prediction Window (180 days)

Figure 8: Experimental setting of CHF early prediction.

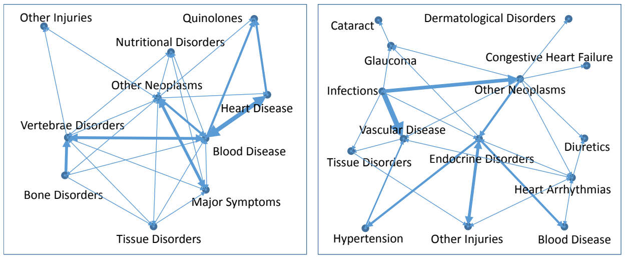
matched controls. For every patient, we set an operation criterion date, which is the CHF confirmation date for case patients, the last day in our database for control patients. We then trace back from the operation criterion date, hold off the records with in the prediction window (180 days), and use the records in observation window (360 days) for analysis. The graphical illustration of such setting is in Figure 8.



(a) Case instance.

(b) Control instance.

Figure 9: Temporal graph examples of a case and control patient in hospitalization prediction data.



(a) Case instance.

(b) Control instance.

Figure 10: Temporal graph examples of a case and control patient in CHF prediction data.

In our experiments, the temporal graphs are constructed from the patient EHRs in observation window. Figure 9 and Figure 10 show example temporal graphs of case and control patients for both data sets, where similar as in Figure 2, we use thicker edges to denote stronger weights (which suggests shorter intervals). We tested the different strategies introduced in Section 3 to learn the temporal phenotypes. The composition coefficients for every patient will be used as their vector representations for the prediction task. For comparison purpose, we also implemented the following baselines:

Aggregated Vector Representation (AVR) This method just counts the frequency of every medical event (diagnosis or medications) in each patient's EHR sequence. Each patient will be represented by a vector with the size equal to the number of distinct medical events. Those counts will be the values on the vector in their corresponding dimensions.

Bag-of-Pattern in Sequences (BPS) This method runs a standard sequential pattern mining algorithm to detect frequent patterns from those EHR sequences, and then combine all frequent patterns to form a pattern repository. Every patient will be represented as a vector with dimensionality equal to the size of the pattern repository. The value on a specific dimension will be the frequency that pattern appeared in the EHR sequence of the corresponding patient.

Metric	Graph	Temporal Phenotyping			
		Un	Sim	Logit	Hinge
AUC	0.78±0.05	0.82±0.02	0.79±0.02	**0.84±0.01**	0.81±0.02
AUPR	0.64±0.04	0.71±0.02	0.72±0.05	**0.76±0.01**	0.74±0.03
ACC	0.87±0.08	0.87±0.01	0.89±0.04	0.89±0.02	**0.89±0.01**

Table 1: The classification performance over 10-fold cross validation on synthetic data.

Data	Metric	AVR	BPS	TES	Temporal Phenotyping			
					Un	Sim	Logit	Hinge
CHF	AUC	0.70±0.03	0.69±0.04	0.67±0.02	0.71±0.02	0.69±0.03	**0.72±0.01**	0.72±0.04
	APR	0.41±0.05	0.52±0.06	0.37±0.04	0.62±0.01	0.60±0.04	**0.65±0.01**	0.62±0.03
	ACC	0.76±0.02	0.77±0.08	0.77±0.02	0.77±0.02	0.78±0.02	0.79±0.01	**0.80±0.04**
Hospitalization	AUC	0.56±0.11	0.67±0.05	0.65±0.06	0.73±0.08	0.71±0.10	**0.73±0.06**	0.69±0.10
	APR	0.32±0.09	0.58±0.13	0.38±0.07	0.64±0.04	0.65±0.15	**0.67±0.12**	0.64±0.16
	ACC	0.65±0.11	0.75±0.05	0.73±0.08	0.76±0.07	**0.80±0.07**	0.79±0.04	0.77±0.05

Table 2: The classification performance over 10-fold cross validation on two real-world data sets.

Temporal Event Signatures (TES) This method implements the temporal signature mining algorithm proposed in [28], which identifies the temporal patterns in patient EHRs via a constrained optimization procedure. The patients will still be represented by the bag-of-pattern representation as in the BPS method.

After the vector based representation for every patient is derived, we then adopt Support Vector Machine (SVM) to perform prediction. Similar as in the study on synthetic data, the classification performance is measured by AUC, AUPR, and ACC, and these measures are averaged over 10-fold cross validation. As there are some parameters in our methods, we adopt a greedy method to choose the optimum values of them. Basically we first construct the temporal graphs using the unsupervised method, and the locality controlling factor r is tuned with cross validation on the prediction results using the constructed graph. Then the number of basis K is tuned based on the prediction results with unsupervised phenotyping. Finally the tradeoff parameter λ for regularized phenotyping methods is tuned with r and K fixed. In both studies, we set $\Delta = 90$, i.e., 3 months, as both are chronic disease scenarios. In the following, we document details on choosing r and K.

4.2.1 Tuning Locality Scaling Parameter r

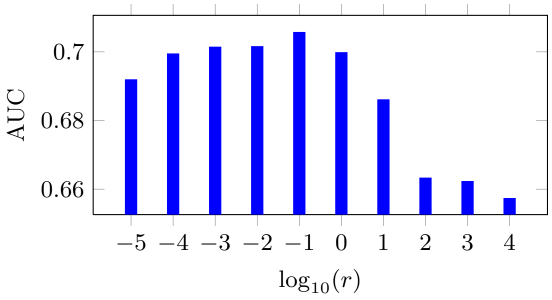


Figure 11: The AUC of graph features with different bandwidth r. Averaged with 10 runs of 10-fold cross validation.

In order to tune the locality scaling parameter r, we first stretch the graph adjacency matrix into a long vector to represent the patients, then we perform prediction with 10-fold cross validation on each $r = r = 10^{[-5:1:5)}$ and select the best. The results are shown in Figure 11, from which we can see $r = 0.1$ gives the best performance AUC=0.705.

4.2.2 Tuning the Number of Phenotypes K

With r fixed to 0.1, the next step is to select the optimal number of phenotypes. We vary the number of basis K from 10 to 500, and perform unsupervised phenotyping as in Section 3.2. The composition coefficients for every patient will be used for prediction and we show the AUC averaged over 10-fold cross validation in Figure 12 with different K values. From the figure we can see that $K = 50$ gives the best AUC=0.717. One phenomenon we can observe from the Figure is that the AUC cannot always increase when we increase the number of phenotypes.

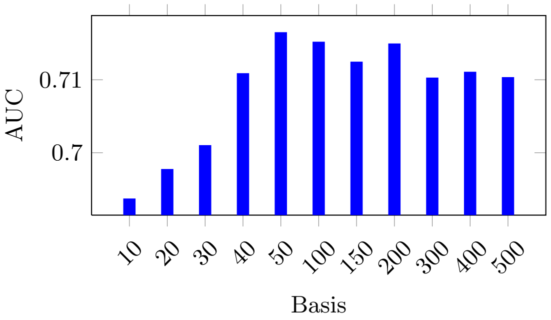
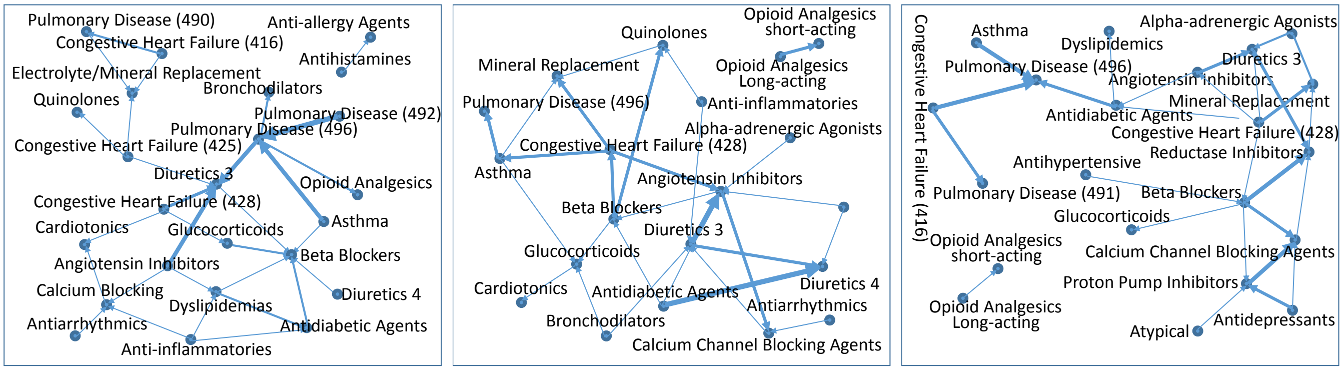


Figure 12: The AUC of phenotyping representations with different number of basis. Averaged with 10 runs of 10-fold cross validation.

4.2.3 Results Summary and Discussion

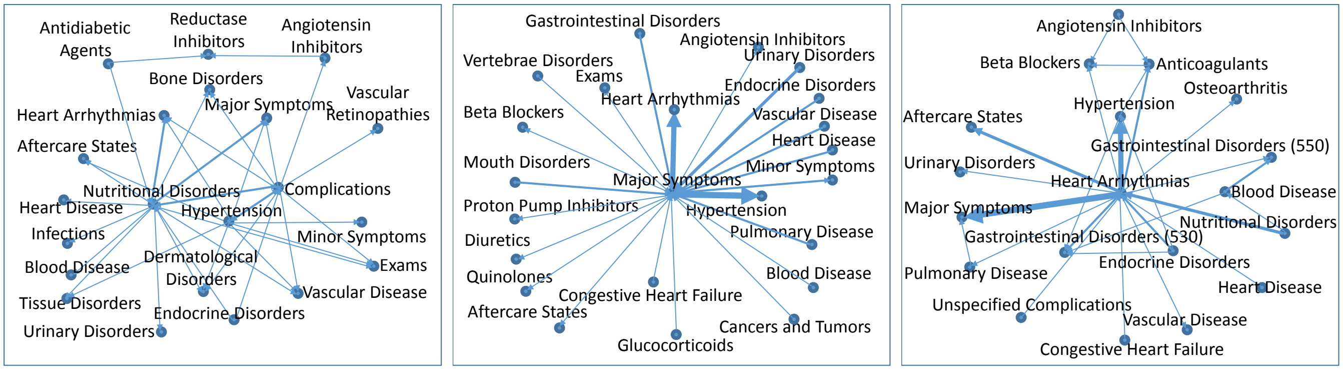
A summary with the quantitative results with parameters chosen in the ways described above is provided in Table 2. From the table we can observe that:

- Representation with our proposed methods can achieve better prediction performance compared to those baselines, which suggests the effectiveness of the proposed graphical scheme.



(a) Phenotype I (b) Phenotype II (c) Phenotype III

Figure 13: Example of temporal phenotypes of hospitalization prediction data. The number following a drug name indicates the strength of the drug, i.e., it is used to treat CHF of which stage. The three digits in the parentheses correspond to the first three digits of ICD-9.



(a) Phenotype I (b) Phenotype II (c) Phenotype III

Figure 14: Example of temporal graph basis of CHF prediction data. The three digits in the parentheses correspond to the first three digits of ICD-9.

- Regularized phenotyping generally produces better results because they utilize supervision information. This observation is also consistent with what we had on the synthetic data.

In addition to the quantitative results, we also present three phenotypes with largest magnitude of the average composition coefficients for each data set in Figure 13 and 14 respectively. In Figure 13 of all three phenotypes we can clearly observe: the drug hubs for treating CHF such as diuretics, beta blockers, ACE inhibitors; drug hubs for treating COPD such as glucocorticoids and bronchodilators; disease hubs that is related to CHF, such as CHF and dyslipidemia; as well as COPD disease hubs including pulmonary disease and asthma. One interesting observation is that on both phenotype II and III, there is an isolated pattern from opioid analgesics long acting to short acting. Opioid analgesics is used to relieve severe pain. The transition from long acting to short acting may suggest the patients' condition deteriorates. In Figure 14 phenotype I we can see three hubs in the middle, nutritional disorders, hypertension and complications. Hypertension is highly correlated with CHF [14], and nutrition disorders may also lead to CHF [27]. Also CHF is a complicated chronic disease and it may cause severe complications. Phenotype II is a single-hub structure with major symptoms in the center, which suggests the clinical pathway starts from some checking (major symptoms found). One interesting finding is that the transition from major symptoms to hypertension and heart arrhythmias are shorter than other diseases and drugs, and both of them are high risk factors for CHF (for the role of heart arrhythmia one can refer to [3]). Phenotype III is with heart arrhythmia in the center, other disorders distributed around it. Thus this represents a clinical pathway originated from heart arrhythmia, which is highly correlated with CHF.

5. CONCLUSION

In this paper, we proposed a novel graph based representation for patient EHRs, which encodes distinct medical events as well as their temporal relationships. Compared to traditional sequence and matrix based representations, graphs are more compact and intuitive. We presented several approaches to identify interesting temporal phenotypes based on such graph based representation, and validated their effectiveness on both synthetic and real-world data sets.

Acknowledgements

This research was partially supported by National Science Foundation (CCF-1018151). Also, it was supported in part by Natural Science Foundation of China (71028002).

References

[1] Data driven healthcare. *MIT Technology Review Business Report*, 117(5):1–19, 2014.

[2] Shiyu Chang, Charu C Aggarwal, and Thomas S Huang. Learning local semantic distances with limited supervision. In *ICDM*, 2014.

[3] JohnW Dean and MaxJ Lab. Arrhythmia in heart failure: role of mechanically induced changes in electrophysiology. *The Lancet*, 333(8650), 1989.

[4] David Gotz, Fei Wang, and Adam Perer. A methodology for interactive mining and visual analysis of clinical event patterns using electronic health record data. *Journal of biomedical informatics*, 48, 2014.

[5] Joyce C Ho, Joydeep Ghosh, Steve R Steinhubl, Walter F Stewart, Joshua C Denny, Bradley A Malin, and Jimeng Sun. Limestone: High-throughput candidate phenotype generation via tensor factorization. *Journal of biomedical informatics*, 52: 199–211, 2014.

[6] Joyce C Ho, Joydeep Ghosh, and Jimeng Sun. Marble: high-throughput phenotyping from electronic health records via sparse nonnegative tensor factorization. In *KDD*, 2014.

[7] George Hripcsak and David J Albers. Next-generation phenotyping of electronic health records. *Journal of the American Medical Informatics Association*, 20(1): 117–121, 2013.

[8] Peter B Jensen, Lars J Jensen, and Søren Brunak. Mining electronic health records: towards better research applications and clinical care. *Nature Reviews Genetics*, 13(6):395–405, 2012.

[9] David Kale, Zhengping Che, and Yan Liu. Computational discovery of physiomes in critically ill children using deep learning. In *Workshop DMMI in AMIA*, 2014.

[10] Eamonn Keogh, Kaushik Chakrabarti, Sharad Mehrotra, and Michael Pazzani. Locally adaptive dimensionality reduction for indexing large time series databases. In *SIGMOD*, 2001.

[11] Eamonn Keogh, Kaushik Chakrabarti, Michael Pazzani, and Sharad Mehrotra. Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and information Systems*, 3(3): 263–286, 2001.

[12] Eamonn Keogh, Selina Chu, David Hart, and Michael Pazzani. An online algorithm for segmenting time series. In *ICDM*, 2001.

[13] Thomas A Lasko, Joshua C Denny, and Mia A Levy. Computational phenotype discovery using unsupervised feature learning over noisy, sparse, and irregular clinical data. *PloS one*, 8(6):e66341, 2013.

[14] Daniel Levy, Martin G Larson, Ramachandran S Vasan, William B Kannel, and Kalon KL Ho. The progression from hypertension to congestive heart failure. *Jama*, 275(20):1557–1562, 1996.

[15] Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.

[16] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. A symbolic representation of time series, with implications for streaming algorithms. In *SIGMOD workshop on Research issues in DMKD*, 2003.

[17] Chuanren Liu, Kai Zhang, Hui Xiong, Geoff Jiang, and Qiang Yang. Temporal skeletonization on sequential data: Patterns, categorization, and visualization. In *KDD*, 2014.

[18] Laura B. Madsen. *Data-Driven Healthcare: How Analytics and BI are Transforming the Industry.* Wiley, 2014.

[19] Fabian Mörchen and Dmitriy Fradkin. Robust mining of time intervals with semi-interval partial order patterns. In *SDM*, 2010.

[20] Fabian Mörchen and Alfred Ultsch. Efficient mining of understandable patterns from multivariate interval time series. *Data Mining and Knowledge Discovery*, 15 (2):181–215, 2007.

[21] Robert Moskovitch and Yuval Shahar. Medical temporal-knowledge discovery via temporal abstraction. In *AMIA*, 2009.

[22] Jyotishman Pathak, Abel N Kho, and Joshua C Denny. Electronic health records-driven phenotyping: challenges, recent advances, and perspectives. *Journal of the American Medical Informatics Association*, 20 (e2):e206–e211, 2013.

[23] Adam Perer and Fei Wang. Frequence: interactive mining and visualization of temporal frequent event sequences. In *IUI*, 2014.

[24] Yuval Shahar and Mark A Musen. Knowledge-based temporal abstraction in clinical domains. *Artificial intelligence in medicine*, 8(3):267–298, 1996.

[25] Michael Stacey and Carolyn McGregor. Temporal abstraction in intelligent clinical data analysis: A survey. *Artificial intelligence in medicine*, 39(1), 2007.

[26] Gregor Stiglic, Nigam H. Shah, Niels Peek, and Fei Wang. Workshop at amia on data mining for medical informatics: Electronic phenotyping. Nov 15, 2014.

[27] Stephan von Haehling, Wolfram Doehner, and Stefan D Anker. Nutrition, metabolism, and the complex pathophysiology of cachexia in chronic heart failure. *Cardiovascular research*, 73(2):298–309, 2007.

[28] Fei Wang, Noah Lee, Jianying Hu, Jimeng Sun, Shahram Ebadollahi, and Andrew F Laine. A framework for mining signatures from event sequences and its applications in healthcare data. *IEEE TPAMI on*, 35(2):272–285, 2013.

[29] Jiayu Zhou, Fei Wang, Jianying Hu, and Jieping Ye. From micro to macro: Data driven phenotyping by densification of longitudinal electronic medical records. In *KDD*, 2014.