Temporal Probabilistic Profiles for Sepsis Prediction in the ICU

Eitam Sheetrit*§
Ben-Gurion University
eitams@bgu.ac.il

Nir Nissim†‡§
Ben-Gurion University
nirni@bgu.ac.il

Denis Klimov*
Ben-Gurion University
klimov@post.bgu.ac.il

Yuval Shahar*
Ben-Gurion University
yshahar@bgu.ac.il

ABSTRACT

Sepsis is a condition caused by the body's overwhelming and life-threatening response to infection, which can lead to tissue damage, organ failure, and finally death. Today, sepsis is one of the leading causes of mortality among populations in *intensive care units* (ICUs). Sepsis is difficult to predict, diagnose, and treat, as it involves analyzing different sets of multivariate time-series, usually with problems of missing data, different sampling frequencies, and random noise. Here, we propose a new dynamic-behavior-based model, which we call a *Temporal Probabilistic proFile* (TPF), for classification and prediction tasks of multivariate time series. In the TPF method, the raw, time-stamped data are first abstracted into a series of higher-level, meaningful concepts, which hold over intervals characterizing time periods. We then discover frequently repeating temporal patterns within the data. Using the discovered patterns, we create a probabilistic distribution of the temporal patterns of the overall entity population, of each target class in it, and of each entity. We then exploit TPFs as meta-features to classify the time series of new entities, or to predict their outcome, by measuring their TPF distance, either to the aggregated TPF of each class, or to the individual TPFs of each of the entities, using negative cross entropy. Our experimental results on a large benchmark clinical data set show that TPFs improve sepsis prediction capabilities, and perform better than other machine learning approaches.

KEYWORDS

multivariate time-series, temporal abstraction, sepsis prediction, interval-based temporal patterns.

ACM Reference Format:
Eitam Sheetrit, Nir Nissim, Denis Klimov, and Yuval Shahar. 2019. Temporal Probabilistic Profiles for Sepsis Prediction in the ICU. In *The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '19), August 4–8, 2019, Anchorage, AK, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3292500.3330747

*Department of Software and Information Systems Engineering, Ben-Gurion University
†Department of Industrial Engineering and Management, Ben-Gurion University.
‡Department of Neurobiology, Stanford University.
§Malware-Lab, Cyber-Security Research Center, Ben-Gurion University.

KDD '19, August 4–8, 2019, Anchorage, AK, USA

ACM ISBN 978-1-4503-6201-6/19/08…$15.00
https://doi.org/10.1145/3292500.3330747

1 INTRODUCTION

Sepsis is a life-threatening condition that arises when the body's response to infection injures its own tissues and organs [31]. It may cause a complex variety of physiological, pathological, and biochemical abnormalities, which can lead to organ failure and death. Common signs and symptoms include fever, increased heart and breathing rate, and confusion. Patients who develop sepsis have an increased risk of complications and death, and face higher health care costs and longer hospitalization. Sepsis is a major public health concern, accounting for more than $23.6 billion (6.2%) of total US hospital costs in 2013 [34], and it is one of the leading causes of mortality among patients in intensive care units (ICUs). Today, septic patients are responsible for 10% of the ICU admissions and for occupying approximately 25% of the ICU beds in US hospitals [34].

Sepsis is usually treated with antibiotics and intravenous fluids. Early diagnosis is crucial to proper sepsis management. After the onset of sepsis, the effectiveness of antibiotic treatment rapidly decreases. Research shows that for each one-hour delay in the administration of antibiotic treatment in a case of sepsis, the mortality rate increases by 7% [12]. Early and accurate prediction of the onset of sepsis could facilitate effective and targeted treatment, which could, in turn, reduce the patient death rate and lower the risk of organ damage. Because of the heterogeneous nature of possible infections that can cause sepsis, and the need to process and inspect an enormous volume of time-stamped raw measurements generated by many data sources for each patient, sepsis is difficult for physicians to recognize.

Time series and more generally, time-oriented temporal data, are sequences of discrete or continuous real-valued elements collected over time, such as sensor readings, network monitoring, stock market data, and patient data stored in medical records. The entity (e.g., a computer, a mobile device, a patient) is sampled continuously at fixed or varying time periods. The data may later be represented as time points for some variables (e.g., a heart-rate of 65/min at 17:50:00 on 01/03/17), or as time intervals (e.g., a fever lasting from 10/05/17 to 12/05/17). In most cases, the data collected for each entity are multivariate, i.e., the readings had originated in multiple sensors, which are not necessarily synchronized in any fashion. Basing the analysis on time-*intervals*, rather than on the original time *points* of the raw, time-stamped data, often reveals new relations

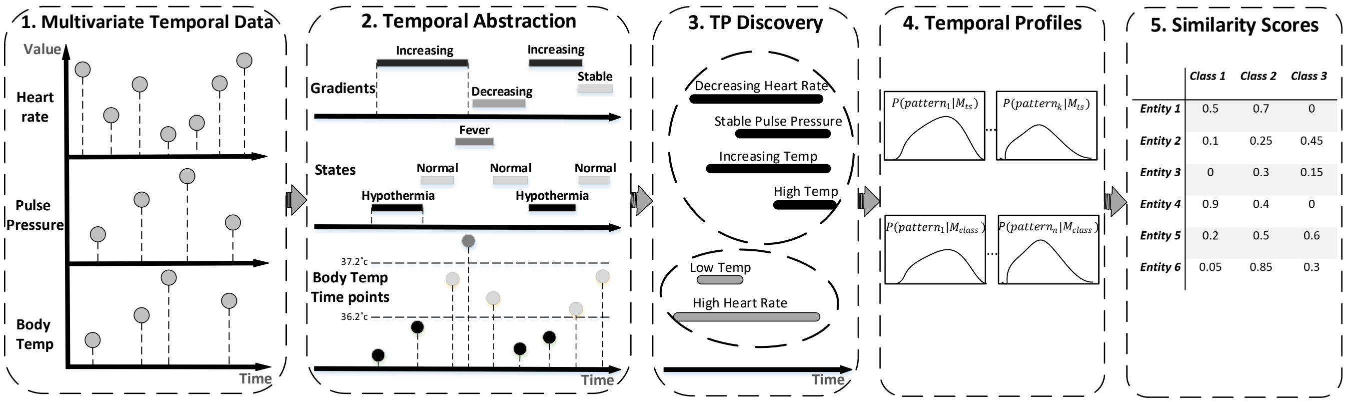

Figure 1: Overview of the temporal-probabilistic profiles-based classification and prediction method.

and patterns within the temporal dimension, since interval-based events may occur concurrently and may have several temporal relations (e.g., Overlap). Furthermore, using abstract concepts (e.g., *a gradual decrease in the performance of the server* over two days or *moderate anemia* for six weeks), rather than only the original raw data (e.g., response times, or raw Hemoglobin values), has many benefits. The abstract concepts are also often, though not necessarily, interval-based. Examples of benefits include reducing random noise in the data, avoiding problems resulting from sampling the data at different frequencies and at various temporal granularities, summarizing large numbers of data in a succinct fashion [7], and helping algorithms in coping with missing data [19, 30]. To support such a meaningful analysis of temporal data, a pre-processing stage, called *temporal abstraction* (TA), is often useful. Such a process summarizes and interprets the raw, point-based, time-oriented data (e.g., a series of time-stamped body temperature values) into interval-based abstractions (e.g., a period of two days of high fever).

The benefits of using TA for classification and prediction tasks were demonstrated in multiple medical domains [2, 10, 22], as well as in the domain of cyber security [23, 24]. Once a series of symbolic intervals have been created, a process of frequent temporal-patterns discovery can start, by searching for repeating patterns of temporal relations (e.g., Before, Overlaps) among the symbolic time-intervals, which satisfy some support thresholds (i.e., minimal frequency) over the set of all of the longitudinal records.

In this paper, we present a new approach to leverage temporal abstraction and time-interval mining, which we call *Temporal Probabilistic Profiles*, to improve prediction in general, and sepsis prediction in the ICU context in particular, when provided with multivariate time-series data, especially when dealing with the problems, common in clinical medicine, of missing data, different sampling frequencies, and random noise in cases of high-dimensional data. Figure 1 illustrates our proposed approach. First, we acquire the raw, time-stamped multivariate temporal data collected by various sensors (1); then, we abstract the raw time-stamped data into several interval-based abstract concepts, each belonging to one of several abstraction types, as we explain in Section 2.1 (2). In the third step, we discover, within the interval-based abstractions of the data, relatively frequent (i.e., discovered at a frequency that is above some predetermined minimal-support threshold) interval-based temporal patterns (3). Then, we create *Temporal Probabilistic Profiles* based on the distribution of the discovered frequent patterns (4). Finally, we leverage the temporal probabilistic profiles as meta-features, to create similarity scores for the purposes of predicting sepsis (5).

Our contributions in this study are as follows: (1) Introducing a new method for temporal-pattern-based classification and prediction that exploits a new similarity measure, based on a new representation scheme, *Temporal-Probabilistic Profiles*, which is inspired by measures used in the Information Retrieval area, and in particular, by language modeling. (2) Exploiting an additional knowledge-based temporal-abstraction type that can be created within the TA phase, in addition to existing ones, to better capture the temporal behavior of the data. (3) Demonstrating the effectiveness of our proposed method, in the challenging, critical, ICU sepsis-prediction task.

The rest of the paper is structured as follows. The next section provides background on temporal abstraction and on time-interval mining. Section 3 presents our novel prediction method, including the use of new temporal abstraction type and features. Section 4 presents an experimental evaluation in the ICU domain. Section 5 discuss about insights from obtained results. Section 6 reviews related work, before Section 7 concludes the paper.

2 BACKGROUND

2.1 Temporal Abstraction

Temporal abstraction (TA) is the task of representing time-stamped raw data as a set of time intervals, typically at a higher level of abstraction [4, 25, 28]. Given a set of time-stamped data, external events, and sometimes also abstraction goals (which provide additional contexts for the TA process), TA produces a set of interval-based abstractions of the data that represent past and present states and trends that are relevant for the given set of goals. The interval-based abstractions are sensitive to the temporal *context* in which each datum appears [26], such as a blood-glucose measurement of an infant or of an adult person, or that was taken within the temporal span of the effect of a short-acting Insulin administration; or a suspicious network socket opening within the first 30 seconds after installing a new operating system. TA, which in most cases includes some form of interpolation [27], may be helpful when dealing with data that were sampled at different frequencies or that lack some missing values, or when trying to mine time-points and time-intervals together, due to its smoothing effect on the generated abstractions. Temporal abstraction techniques usually include a discretization module as a pre-processing step, which discretizes raw data into a small number of possible values, by determining cut-offs values. The discretization can be performed either by automatic unsupervised techniques [6, 14, 16], supervised methods [17], or by using knowledge acquired from a domain expert [25]. Figure 2 describes an example of a temporal abstraction of a series of raw time-point data for a patient's body temperature. The data in this case are abstracted according to their values into three interval-based abstractions: State (a classification of the range of value), Gradient (the sign of the first derivative), and Trend abstractions (see details later on in the text).

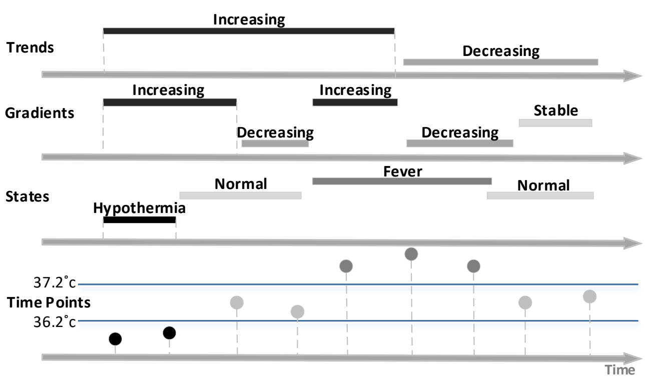

Figure 2: An example of a temporal abstraction of a patient's body temperature into three interval-based abstractions

2.2 Mining Time-Intervals

Time-interval mining is the task of finding frequent relations between time-intervals that exist in the time-oriented data. Most time-interval mining methods rely on Allen's temporal relations [1] to represent the temporal logical relations among the time intervals. Hoppner [8] introduced a method to mine rules in symbolic interval sequences using Allen's temporal relations, and a non-ambiguous representation of time-interval patterns, in order to represent all of the pairwise relations within a k-intervals pattern. Papapetrou [20] presented a time interval mining method while using only five temporal relations, and introduced an epsilon temporal-distance threshold to make the temporal relations more flexible. Moskovitch and Shahar proposed an efficient method, called the *KarmaLego framework* [19], which exploits the transitivity of temporal relations to generate candidates efficiently, while also holding direct pointers to the relevant patterns at each node of the evolving temporal-patterns tree.

The use of *interestingness* (*feature-representation*) measures for the discovered temporal patterns, also known as quantitative measures, plays an important role in their exploitation for classification and prediction tasks. Many *interestingness* measures have been proposed and studied [18–20], each of them capturing different characteristics. The most common one is *vertical support* - Given a database of $|E|$ distinct entities, the *vertical support* of a temporal pattern tp is denoted by the cardinality of the set E^{tp} of distinct entities in which tp holds at least once, divided by the total number of entities $|E|$. When a temporal pattern has vertical support above a minimal predefined threshold, it is referred to as *frequent*. Since a temporal pattern can be discovered multiple times within a single entity, we can also define, following Moskovitch and Shahar [19], another type of support, called *horizontal support*. The *horizontal support* of a temporal pattern tp for a certain entity e_i is the number of times that the particular temporal pattern tp was found in e_i.

Recently, several studies proposed using the discovered temporal patterns as features, for classification tasks of multivariate time series. *IEClassifier* method (Patel et al.) [21] selects, using information-gain, a subset of temporal patterns that is able to discriminate one class from the other. Then, for classifying an unknown input event list *I*, it assign the class with the largest number of discriminating patterns, or the class with the highest confidence patterns appearing in *I*. Batal et al. [2] presented an apriori approach, called *STF-Mine*, for the discovery of temporal patterns, and used the χ^2 (chi-square) measure to select the most discriminating patterns, which will later be used as features in standard classification algorithms (*SVM*, *C4.5*).

Although several studies have proposed using time interval patterns for classification, no study has investigated the effect of using multiple different abstraction types [e.g., States, Gradients, Trends] as the basic components of the patterns, or the effect of using *the entire [frequent] temporal patterns distribution* as a new [meta-]feature, on the classification performance. In the current study, however, we examine these aspects in detail, and they represent several of the contributions of our work.

3 TEMPORAL PROBABILISTIC PROFILES

In this section, we present our new approach, called *Temporal Probabilistic Profiles*, inspired by techniques from Information Retrieval in general, and by language models in particular [32], for the task of classification and prediction of multivariate time-series data. The basic steps of our approach are illustrated in Figure 1. Below, we provide further details regarding the concepts underlying our overall methodology, as well as regarding the methods' formulation and a pseudo code of the relevant algorithms.

3.1 Problem Statement

Given a set of k different classes, each contains n_k unique entities with their multivariate time series data, and another unclassified entity, we wish to classify the unknown entity, using its multivariate time series data, to the class to which it is most likely to belong, i.e., which is most likely to have generated the given time-series data. To tackle this task, we introduce a new term, called *Temporal Probabilistic proFile* (TPF), on which we now elaborate.

3.2 Method Description

A *Temporal Probabilistic proFile* (TPF) is a probability distribution that aims to capture the dynamic behavior of the entity generating the multivariate time series. It characterizes a time series by assigning a probability to the occurrence of each frequent temporal pattern in it, determining how likely a given temporal pattern, or a series of patterns, is to appear in the profile, given an entity that generates the temporal patterns. The approach is inspired by the use of language models in information retrieval [32], where according to the basic idea, a query q is "generated" by a probabilistic model based on a document d. This approach is composed of four stages (as presented in Algorithm 1): Given a multivariate time series data, of k different classes, each contains n_k unique entities, and another unclassified set of entities with their multivariate time series data:

(1) Abstract a multivariate time series data set into a set of intervals using three abstraction types.
(2) Generate a collection of all frequent interval-based temporal patterns appearing in the time series collection, with their selected feature-representation method.
(3) Use one of two strategies: (1) A ***Class-model*** strategy - Create an aggregate TPF for the time series data of all entities that belong to the same class c_i. The TPF of c_i will be the set of all multisets over the frequent temporal patterns discovered in that class. (2) A ***TopK*** strategy: Create a TPF for each of the entities, or instances, appearing in each class. The TPF will be the set of all multisets of frequent temporal patterns discovered within the data of that entity (instance).
(4) Find the similarity between the TPF of a new time series instance, which represents an unknown given entity (instance), and a set of existing TPFs, representing either (1) an aggregation of the various classes, or (2) the full set of [labeled] individual entities within the training set. Determine the class in which the entity is most likely to be a member. To this end, we use the *Kullback-Leibler* (KL) divergence score [11], or more precisely, the negative cross entropy, and rank the various classes accordingly, where we assume that the known class-model is the true distribution, and we want to assess, how likely is the unknown entity to be a member of that known class distribution. In the case of (2), we can select the top k most similar entities, and perform a voting to determine the classified class. Moreover, using KL-Divergence (or its symmetrized version - Jensen-Shannon divergence [13]) also allows us to measure the degree of similarity between the TPFs of two time-series, for instance in the case of matching tasks.

Algorithm 1: TPF Classification

Inputs : k different classes, each contains n_k unique entities, and m unknown entities; MTS - raw multivariate time series data of each entity; minimal support

Outputs : Classification of one out of k classes for each unknown entity

```
1:  Classification[] ← ∅, SimilarityList[][] ← ∅
    /* lines 2-9 are performed once, as a pre-processing step, for labeled entities */
2:  AMTS ← CreateTemporalAbstraction(MTS)
3:  TP_set ← TemporalPatternsDiscovery(AMTS, MinSupport)
4:  foreach class c_i do
5:      TPF_c[c_i] ← CreateTPF(TP_set, c_i)
6:  end
7:  foreach entity e_i do
8:      TPF_e[e_i] ← CreateTPF(TP_set, e_i)
9:  end
    /* the following lines are performed at runtime */
10: foreach unknown entity e_j do
11:     AMTS_{e_j} ← CreateTemporalAbstraction(MTS[e_j])
12:     TP_{e_j} ← DetectDiscoveredPatterns(AMTS_{e_j}, TP_set)
13:     TPF_{e_j} ← CreateTPF(TP_{e_j}, TP_set)
14:     if Class-Model strategy is chosen then
15:         MaxSimilarity ← 0, Class ← ∅
16:         foreach class c_i do
17:             Similarity ← Sim(TPF_{e_j} | TPF_c[c_i])
18:             if Similarity > MaxSimilarity then
19:                 MaxSimilarity ← Similarity
20:                 Class ← c_i
21:             end
22:         end
23:         Classification[e_j] ← Class
24:     else if TopK strategy is chosen then
25:         foreach entity e_i do
26:             SimilarityList[e_j][e_i] ← Sim(TPF_{e_j} | TPF_e[e_i])
27:         end
28:         Choose Top K from SimilarityList[e_j]
29:         Classification[e_j] ← dominant class in Top K entities
30: end
31: return Classification
```

3.3 Temporal-Abstraction Generation

At the temporal abstraction phase, we generate three types of abstract concepts, or three abstraction types, to capture the various aspects of the dynamic temporal behavior of each entity. In our study, we had applied the KBTA method, and refer the reader to Shahar's original description [25] for formal definition of the basic abstraction types and their generation methods. In particular, we use the *State* abstraction, which classifies the values of one or more contemporaneous [raw or abstract] concepts into the value of an abstract concept (e.g., Low, Normal, or High Activity Level, given several of its raw-data components), and the *Gradient* abstraction, which classifies the values of a concept that holds over some time period into a set of change directions (e.g., Decreasing, Stable, Increasing). With the help of our domain expert, we identified 26 clinical concepts related to sepsis and then established knowledge-based state, gradient, and trend abstraction definitions. The definitions were used for the abstraction of the raw data into more meaningful concepts. Table 1 describes the knowledge base used for the process of abstraction: (1) *Clinical concept* - the medical measurement's name. The first 17 concepts are laboratory blood tests, and the next nine are vital signs and bedside measurements. (2) *"Normal" state values* - the recommended values for a healthy patient. Each concept is abstracted into three states, based on those values: Normal (the values mentioned in the table), Low (values that are lower than normal), and High (values higher than normal).

Table 1: Sepsis related knowledge base

Clinical Concept	"Normal" State Values	Gradient/Trend sigValues	Trend tStable
Albumin	3.4-5.4 g/dL	$\Delta > 0.5$	36 hours
Bilirubin	0.2-1.2 mg/dL	$\Delta > 0.5$	36 hours
Chloride	96-106 mEq/L	$\Delta > 5$	36 hours
Creatinine	0.6-1.3 mg/dL	$\Delta > 0.2$	36 hours
Fibrinoge	200-400 mg/dL	$\Delta > 50$	36 hours
Glucose	70-100 mg/dL	$\Delta > 10$	36 hours
Hemoglobin	11-18 g/dL	$\Delta > 2$	36 hours
Lactate	0.5-2.2 mmol/L	$\Delta > 1$	36 hours
PCO2	38-42 mm Hg	$\Delta > 2$	36 hours
PH	7.34-7.45 pH	$\Delta > 0.05$	36 hours
Phosphate	2.4-4.1 mg/dL	$\Delta > 0.5$	36 hours
PLT	150-400 $x10^9$/L	$\Delta > 50$	36 hours
PO2	75-100 torr	$\Delta > 10$	36 hours
Urea	10-20 mg/dL	$\Delta > 5$	36 hours
Sodium	135-145 mEq/L	$\Delta > 5$	36 hours
TCO2	22-28 mmol/l	$\Delta > 2$	36 hours
WBC	4.5-10 $x10^9$/L	$\Delta > 1$	36 hours
Body Temp	$36.2 - 37.2°$C	$\Delta > 0.5$	2 hours
Glasgow CS *	8-12	$\Delta > 2$	2 hours
Diastolic Pressure	70-90 mmHg	$\Delta > 10$	1 hour
Systolic Pressure	110-140 mmHg	$\Delta > 10$	1 hour
Mean Pressure	65-80	$\Delta > 5$	1 hour
Heart-Rate	60-80 bpm	$\Delta > 10$	1 hour
Minute-ventilation	5.4-11 L/min	$\Delta > 0.5$	1 hour
Pulse-Pressure	35-45 mmHg	$\Delta > 5$	1 hour
Respiratory-Rate	7-14 breath/min	$\Delta > 3$	1 hour

*Glasgow Coma Scale states are mild, moderate and severe

(3) *sigValue* - the minimal required difference between values of two data points, in order to recognize a change. Those values are used in the gradient and trend abstraction process, in order to determine if the values are ’Stable’ (a change in values that is equal to or smaller than the one mentioned in the table), ’Increasing’ (a positive change larger than the value appearing in the table), and ’Decreasing’ (a negative change with an absolute value larger than the value appearing in the table). (4) *tStable* - the minimal time period that a concept needs to remain without a significant change, in order to be considered as ’Stable’

Here we shall briefly describe the computational principles of the newly proposed *Trend* abstraction type (Algorithm 2), which was found most useful in our study to describe changes in a temporal behavior. We start by partitioning the input time series of a single measure into a set of ordered clusters, or partitions, such that each partition contains only raw time points in which the difference between the timestamps of consecutive points is less than maxGap. Then, for each partition, we search for all local extremum points, and ignore the rest of the points. For each two consecutive extremums, we calculate their value difference in order to determine the abstract Trend characterization of the period between them (Increasing, Decreasing, or Stable).

Algorithm 2: Trend Abstraction Generation

Inputs : $T[(time, value)_i]$, $i = 1...N$ - a set of N temporally-ordered samples of a measure for a specific entity, $sigValue$ - a significant value to recognize a change. $maxGap$ - the maximal time gap between consecutive measurements for interpolation. $tStable$ - the minimal time period that a concept needs to remain without significant change, in order to be tagged as ’stable’.

Outputs : $I[(start, end, value)_j]$, $j = 1...M$ - a set of M trends intervals.

$TimePartition \leftarrow \emptyset; Interval \leftarrow \emptyset; j \leftarrow 1$
$TimePartition_j.add(T_1)$
for $i \leftarrow 1$ *to* $T.Size() - 1$ **do**
  if $T_{i+1}.time - T_i.time < maxGap$ **then**
    $TimePartition_j.add(T_{i+1})$
  else
    $j \leftarrow j + 1$
    $TimePartition_j.add(T_{i+1})$
end
foreach $P \in TimePartition$ **do**
  $E \leftarrow$ FindLocalExtremum(P) ▷ in the usual math sense
  while $i < E.Size() - 1$ **do**
    $timeDiff \leftarrow E_{i+1}.time - E_i.time$
    $valueDiff \leftarrow E_{i+1}.value - E_i.value$
    if $abs(valueDiff) > sigValue$ **then**
      if $valueDiff > 0$ **then**
        $I.add(E_i.time, E_{i+1}.time, \text{‘}inc\text{‘})$
      else
        $I.add(E_i.time, E_{i+1}.time, \text{‘}dec\text{‘})$
    else
      if $timeDiff \geq tStable$ **then**
        $I.add(E_i.time, E_{i+1}.time, \text{‘}stable\text{‘})$
      else
        $I.UpdatePreviousInterval(I_{prev}.end \leftarrow E_{i+1}.time)$ ▷ tStable param smooths the trend
    $i \leftarrow i + 1$
  end
  $Interval.add(I)$
end
return $Interval$

Note that while the *Gradient* concept indicates the direction of change of a raw time-stamped parameter, such as the values of the Blood Pressure, in the case of the medical domain, the direction of changes in the measured concept’s values might alternate at a rather high frequency (a "zig-zag" concept). Thus, we had introduced a complementary Trend abstraction, to smooth the zig-zags and detect also the global change trend, if any, in the concept’s values, by only looking at the extremum data points. The Trend abstraction is thus akin to a "symbolic" linear regression, but exploits some domain- and context-sensitive knowledge, such as the context-sensitive minimal significant change in the concept’s value that can be considered as Decreasing or Increasing, and the tStable parameter, both of which lead to a smoothing of the Trend abstraction's output. Both the Gradient and Trend abstractions are measured on the ordinal symbolic scale of "Decreasing", "Stable" (i.e., no significant change in the concept's value during a particular period), and "Increasing".

3.4 TPF Formal Definition

A temporal pattern tp_i is a discovered temporal pattern, where $i = 1...N$, and N is the total number of discovered patterns in all classes. A set TPS is a set of temporal patterns, while $\mathcal{M}(TPS)$ is the set of all multisets over temporal patterns. The signature of a time series ts, $Sig(ts)$, $ts \in \mathcal{M}(TPS)$, is defined as a multiset of temporal patterns appearing in ts. $ts(tp_i)$ is the set of occurrences of a temporal pattern tp_i in ts. $|ts(tp_i)|$ is the size of the set $ts(tp_i)$, which is the number of occurrences of a temporal pattern tp_i in ts (horizontal support). A Simple frequency model, for the prior probability of a frequent pattern given a time series, using maximum likelihood estimation (MLE), which defines the distribution of temporal patterns within a time series, is given by the equation:

$$P_{MLE}(tp_i|ts) = \frac{|ts(tp_i)|}{\sum_{j=1}^{N} |ts(tp_j)|} = \frac{|ts(tp_i)|}{|Sig(ts)|} \quad (1)$$

The TPF of a multivariate time series ts_j is a probability distribution, and defined by a vector of probabilities for each $tp_i \in ts_j$:

$$TPF(ts_j) = [P_{MLE}(tp_1|ts_j), ..., P_{MLE}(tp_k|ts_j)] \quad (2)$$

As it is quite likely that a temporal pattern will not appear in a certain multivariate time series ts, causing $|ts(tp_i)|$ to be equal to 0, we use a mixture model that mixes the probability from the time series with the frequency of the pattern in the general collection, using Bayesian smoothing with Dirichlet priors:

$$P_{Dir}(tp_i|ts) = \frac{|ts(tp_i)| + \mu P_{MLE}(tp_i|TS_c)}{|Sig(ts)| + \mu} \quad (3)$$

where μ is a free parameter, usually chosen as the mean size (cardinality) of the signature of a time series (in terms of number of temporal patterns it contains), and $P_{MLE}(tp_i|TS_C)$ denotes the distribution of the temporal pattern in the entire time series collection.

Finally, in order to compare the distribution of frequent patterns in a given entity's multivariate time-series with their distribution in each class, or even with other entities, we use the *negative cross entropy* (CE) [11] for ranking purpose, which is designed to measure how likely is an unknown instance's time series TPF to be a member of, or to be generated from, the distribution of a known class's TPF:

$$\begin{gathered} Sim(TPF(ts_j)|TPF(ts_c)) = \\ exp(-CE(P_{MLE}(\cdot|ts_j)||P_{Dir}(\cdot|ts_c))) \end{gathered} \quad (4)$$

where

$$\begin{gathered} -CE(P_{MLE}(\cdot|ts_j)||P_{Dir}(\cdot|ts_c)) = \\ \sum_{tp_i} P_{MLE}(tp_i|ts_j) \ln(P_{Dir}(tp_i|ts_c)) \end{gathered} \quad (5)$$

The obtained ranking scores are then used in the classification and prediction process, according to the selected strategy, Class-model or TopK, as was described in Section 3.2

4 EVALUATION

Our evaluation uses the MIMIC-III dataset [9], which is comprised of de-identified health-related data associated with over 40,000 patients who stayed in critical care units at Beth Israel Deaconess Medical Center between 2001 and 2012. With the help of our domain expert, we identified 26 clinical concepts related to sepsis, which can be divided into two groups: (1) lab tests (i.e., glucose, platelets, etc.); (2) chart items (blood pressure, heart rate, etc.). Then we established knowledge-based state, gradient and trend abstraction definitions, using guidelines which are described in Section 2.1. The definitions were used for the abstraction of the raw data into three states (normal, low or high values), and three gradients and trends (increasing, decreasing or stable values).

Because the precise sepsis onset time is not included in the MIMIC-III database, the first task was to identify patients with sepsis and determine its onset time. Patients with sepsis were identified using the ICD-9 (International Classification of Diseases, Ninth Revision) diagnosis codes (995.91 and 995.92) [31], if they appear in a patient's discharge forms, and whether they received antibiotics during their stay in the ICU. The sepsis onset time was determined as the earliest between being prescribed antibiotics, and the time the qSofa criteria [31] indicated the existence of sepsis.

We examined the data of adult patients (15 years old or older), that stayed in the ICU for at least four days and, in the case of those that developed sepsis, only those patients who developed sepsis on the third day of their stay or later, in order to be able to collect enough data prior sepsis onset. Although every patient in the ICU has a high probability of sepsis when admitted, only about 50% actually develop it [31]. As some ICU patients might carry a hidden infection from another place, we wanted to make sure that they became septic after a period of not having any infection, in order to achieve unbiased assessment for our prediction method. Those conditions resulted in a total of 2,493 patients: 1,034 septic (41%). and 1,459 non-septic patients.

As the purpose of the evaluation was to predict sepsis in ICU, we examine the behavior of Septic patients until the last 12 hours, and 1 hour prior to sepsis onset, compared to non-septic ICU patients during the equivalent time as the control-group, which were carefully chosen by our domain expert, by having similar demographic characteristics and hospitalization reason. While vital signs and bedside measurements in the MIMIC-III database are being recorded every hour (and sometimes even every 15 minutes), laboratory blood tests are less frequent, and being performed only once a day. In order to handle the low frequency, we decided that laboratory tests are gathered starting from the ICU admission time, and the other measurements are gathered only in the 12 hours window. Both gathering processes end at the end of the 12 hours window. We then divided the patients into 8 groups according to their gender and age, as demographic features also affect the behavior of sepsis. Table 2 presents the distributions of patients, where black numbered color indicates septic patients, and gray colored number indicates non-septic patients. The next step is the discovery of temporal patterns hiding in the interval collection. To this end, we use *KarmaLego* [19], as mentioned in Section 2.2. We search for temporal patterns (TP) whose frequency is above 20% support threshold, in every class of patients (septic and non-septic) in each gender

Table 2: Distribution of patients

Gender	Ages 15-50	Ages > 50
Male	**120**/179	**516**/664
Female	**78**/122	**320**/494

and age group separately. For each discovered pattern, we use horizontal support as the value. Each of the following experiments was performed 5 times (folds), and the results were averaged.

4.1 Assessing TPF

We start by conducting statistical analysis on two levels: global, with a two-sample Kolmogorov-Smirnov test, and local, with proportion test [15]. The Two-sample Kolmogorov-Smirnov test quantifies a distance between the empirical distribution functions of two samples. The null distribution of this statistic is calculated under the null hypothesis that the samples are drawn from the same distribution. This test will allow us to investigate whether the two distributions of TPs discovered in the two classes are different. While Kolmogorov-Smirnov captures the whole set of TPs and may be affected by TPs appearing in only one class, the proportion test can test a single TP at a time.

4.2 Different abstraction types

This experiment was designed to investigate the influence of different abstraction types on the prediction capabilities. At the first stage, we used the same abstracted data as described earlier in this section, randomly divided our patients in each class to training set (80%) and test set (20%). Using the training data, we searched for patterns in each class that appeared (at least once) within at least 20% of the training patients belonging to that class. Then, for each test patient, we marked which of the discovered patterns (from both classes according to his age and gender) appeared at. Finally, we used TPF to classify the test data using the training examples. On the second stage, we used data abstracted only to states (without gradient and trend abstractions) and perform the same steps as on the first stage.

4.3 Classification methods comparison

Finally, we investigate and compare the results obtained using our proposed classifier, and results obtained by popular machine learning methods: (1) *Random Forest* and (2) *Recurrent Neural Networks* (RNN). For (1) we used the *STF-Mine* methodology [2], described in Section 2.2, using the same discovered temporal patterns as features. As we are dealing with multivariate raw data with irregular sampling, varying sampling frequencies and missing values, a traditional RNN cannot effectively handle such data [3], so for (2) we decided to use a special version of *RNN* for Multivariate Time Series with Missing Values called GRU-D, as described in [3], using the original multivariate raw data, and patients' demographic features, in order to perform a fair comparison to TPFs. We were assisted in our implementation by the original GRU-D authors.

Table 3: Proportion test results, 1-hour before sepsis onset.

Test Description	%TP with diff. prop.
Septic vs. Non-Septic groups	65%
Within Septic groups	4%
Within Non-septic groups	5%

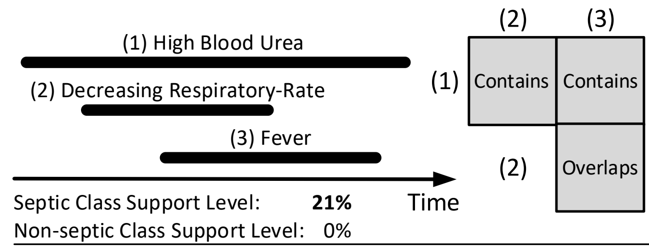

Figure 3: An example of a temporal pattern discovered at the mining stage. The level of support appears for each of the two classes (septic and non-septic patients).

4.4 Results

We started by discovering a total of 145,071 interval-based temporal patterns that appeared at least once within at least 20% of the patients of at least one of the classes. Out of these patterns, 63,596 where exclusively found within the Septic-patients classes, 7,802 where exclusively found within the non-septic classes, and the rest (73,673) appeared within the longitudinal records of both patient classes. Figure 3 presents an example of a temporal pattern discovered exclusively in the septic class, including three time-intervals (two state abstractions and one trend abstraction) and all of the pairwise temporal relations between them. Using the Two-sample Kolmogorov-Smirnov test, we discovered that the distribution of temporal patterns for the two classes in each gender and age group is statistically different with a confidence level of 0.95 ($\alpha = 0.05$).The proportion tests (also with $\alpha = 0.05$) support the results, by presenting a significant difference in the relative amount of different proportions of TPs *between* the two classes in all genders and age groups, in comparison to the relative amount *within* the classes, as presented in Table 3.

We later investigate the influence of different abstraction types on the prediction results. Table 4 presents the results, using data abstracted only to states (State Abstraction), and a combination of different levels of abstractions (State+Gradient+Trend), using both strategies, with different settings for the K parameter. Although the use of state abstraction is more intuitive for physicians, and common in other studies, it is, in fact, the combination of state, gradient, and trend abstraction that yields the best results; thus, in Table 4 we chose to display only those two temporal abstraction combinations. We can clearly see that using all three abstraction types produced the best results, regardless of the chosen strategy. It is worth noticing the influence of K on the prediction results, where $K = 13$ and $K = 17$ yield the best result.

Finally, Figure 4 compares the results of the Class-Model and TopK strategy (with $K = 13$, which yields the best result according

Table 4: Prediction results with different abstraction levels, 1-hour before sepsis onset.

Strategy	State Abstraction		State+Gradient+Trend	
	TPR	FPR	TPR	FPR
TPF Class-Model	0.702	0.006	**0.818**	**0.002**
TPF TopK = 1	0.426	0.005	**0.477**	0.005
TPF TopK = 5	0.640	0.005	**0.714**	0.005
TPF TopK = 9	0.712	0.003	**0.890**	**0**
TPF TopK = 13	0.751	0.003	**0.912**	**0**
TPF TopK = 17	0.824	0	**0.911**	**0**
TPF TopK = 21	0.774	0	0.758	0.002

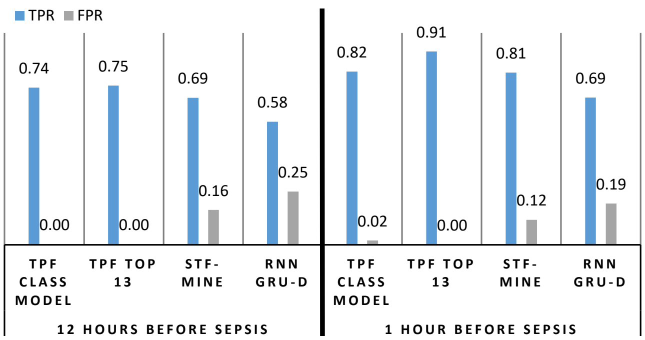

Figure 4: Comparison to other classifiers, on two different time windows.

to the experiment 2), with the results obtained with *STF-Mine* using a subset of the most discriminating patterns out of the entire discovered temporal patterns' set, according to the χ^2 (chi-square) measure, and the results of the recently proposed *RNN GRU-D* using the original multivariate raw data (note that the original RNN, even in its Long Short-Term Memory (LSTM) version, can accept as input several different multivariate time series as vectors, but the temporal relations among them, as well as missing data, are no longer explicit). Overall, our proposed TPF, using TopK strategy, with $k = 13$ achieves the best sepsis prediction result, on each selected time window prior to sepsis onset, outperforms *GRU-D*, and produces better results than *STF-Mine*, and with similar results to the Class-Model on the 12 hours window.

5 DISCUSSION

One of the main problems, when dealing with clinical data, is that the original raw data are usually multivariate, and are sampled irregularly (in time), with various sampling frequencies and missing values. Popular deep learning techniques, such as RNNs, cannot effectively handle such data, as opposed to our proposed TPF method, which uses highly abstract meta-features that represent the entity's full distribution of frequent temporal patterns. Even recent RNN architectures, specifically designed for processing multivariate time-oriented data with missing values, have difficulties catering to the multiple levels of abstractions inherent in complex interval-based temporal-abstraction patterns, as was confirmed by the results.

Moreover, adding to the pre-processing step Gradient and Trend temporal abstractions, which focus on different resolution-level changes in the variables' values, beyond the State abstraction (which focuses on discretization of the values), added considerable classification power to the temporal profile. In addition, the very difference in temporal patterns distribution among the classes, found in the temporal patterns' discovery step, might provide additional insights to domain experts. These insights provide a modicum of explainability to the prediction, an aspect that is often lacking when using deep learning, as effective as it might be.

A note regarding real-time performance: while the *TopK* approach requires comparison against each known instance, the *Class-model* aggregates all instances in each class, and transforms the comparison task into examining the representative distribution of only each *class* in the data, and not each *instance* in these data. When building the model, one needs to go over all instances of that class; but that happens only once, as a pre-processing step. In our current evaluation, the *Class-model* had provided a performance that was very close to the *TopK* approach, with a significantly lower computational effort. Thus, selection of the appropriate method can be made during runtime using external constraints.

Our next objective is to test the system on the real-world retrospective database of our local academic medical center's ICU, working with our local ICU medical expert. If the results are satisfactory, we plan to install a version of the system in the real-time environment of that ICU.

6 RELATED WORK

In several studies, researchers have tried to tackle the problem of septic shock prediction, but relatively few have attempted to predict the more common complication of sepsis. Septic shock is a subset of sepsis in which underlying circulatory and cellular abnormalities are profound enough to substantially increase mortality [31]. Since septic shock is an aggravation of sepsis, detection of sepsis at an early stage is preferred, and might prevent the patient from deteriorating into a septic shock. Most of the proposed methods have been tested using the MIMIC database.

Desautels et al [5] proposed a sepsis predicting method based on a machine learning classification system, called InSight, that uses multivariate combinations of easily obtained patient data (vitals, peripheral capillary oxygen saturation, Glasgow Coma Score, and age), to predict sepsis using the MIMIC-III dataset, limited to ICU patients 15 years or older, who stayed in the ICU for at least seven hours prior getting sepsis. The features used by InSight are the value changes in each of these clinical types of data, between the point we wish to make a prediction, and two hours back. Nonlinear function approximations for each posterior probability of sepsis, was created for the prediction score. In testing, InSight performed as well as, or better than the traditional scoring systems for data collected 1-2 hours prior to the patient reaching a sepsis state, with an average F-score of 0.48.

Other recent work has focused on early prediction of septic shock. A septic shock early warning system (EWS) was developed using multivariate logistic regression on commonly measured clinical variables [29]. Using a dataset with 65 septic shock and 185 sepsis only patients, the model could predict the onset of septic shock one hour in advance with an AUC of 0.928. However, the system used invasively-gathered data and extracted features from the MIMIC waveform data, which provide higher time resolution than data more commonly available in most ICUs. Khoshnevisan et al [33], tried to predict septic-shock using a sequential pattern-based approach, called RTPs. They first selected 17 medical features, in addition to the current medical state, and the location of the patient, as raw data. Then, they abstract the raw data using knowledge-based state abstraction, in order to perform time-intervals mining using two temporal relations: before, and co-occurs. Using a dataset of 3,738 patients: 1,869 positive septic-shock patients, and the same amount of negative ones, they have demonstrated an F-measure of 0.85 in the task of predicting septic-shock 4 hours before onset. Another study performed a recursive partitioning and regression tree (RPART) analysis on 1864 septic patients to identify early predictors from the clinical data of hospitalized non-ICU patients [33]. The model, which required results from 11 routine laboratory tests and basic vital signs, correctly identified 55% of the septic patients.

To conclude, predicting the relatively rare event of septic shock using accessible clinical data typically did not result in very accurate models. Previous research did not usually try to focus on the more difficult, but important, problem of predicting sepsis.

7 CONCLUSIONS

In summary, the proposed TPF model, which uses the full distribution of frequent interval-based temporal-abstraction patterns in each entity, using the TopK strategy, with a relatively small k (13), has achieved the best sepsis prediction result, on each of the time windows prior to the onset of sepsis that were tried (1 hour and 12 hours). The more efficient Class-Model, which represents the patient population using only a single aggregate temporal-pattern distribution, has achieved similar results on the 12 hours window. Both versions had outperformed the recently proposed *RNN GRU-D*, and produced better results than *STF-Mine*.

ACKNOWLEDGMENTS

We thank Dr. Lior Fuchs, from Soroka University Medical Center, who has assisted us in the creation of the Sepsis knowledge base, Dr. Zhengping Che, from the university of Southern California, for the assistance in the implementation of GRU-D from his paper [3], and Mr. Eilon Sheetrit, from the Technion, for the technical guidance on language models in Information Retrieval.

REFERENCES

[1] James F Allen. 1990. Maintaining knowledge about temporal intervals. In *Readings in qualitative reasoning about physical systems*. Elsevier, 361–372.
[2] Iyad Batal, Lucia Sacchi, Riccardo Bellazzi, and Milos Hauskrecht. 2009. Multivariate Time Series Classification with Temporal Abstractions.. In *FLAIRS Conference*.
[3] Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. 2018. Recurrent neural networks for multivariate time series with missing values. *Scientific reports* 8, 1 (2018), 6085.
[4] Carlo Combi, Yuval Shahar, et al. 1997. Temporal reasoning and temporal data maintenance in medicine: issues and challenges. *Computers in biology and medicine* 27, 5 (1997), 353–368.
[5] Thomas Desautels, Jacob Calvert, Jana Hoffman, et al. 2016. Prediction of sepsis in the intensive care unit with minimal electronic health record data: a machine learning approach. *JMIR medical informatics* 4, 3 (2016).
[6] James Dougherty, Ron Kohavi, and Mehran Sahami. 1995. Supervised and unsupervised discretization of continuous features. In *Machine Learning Proceedings 1995*. Elsevier, 194–202.
[7] Ayelet Goldstein and Yuval Shahar. 2016. An automated knowledge-based textual summarization system for longitudinal, multivariate clinical data. *Journal of biomedical informatics* 61 (2016), 159–175.
[8] Frank Höppner. 2001. Learning temporal rules from state sequences. In *IJCAI Workshop on Learning from Temporal and Spatial Data*, Vol. 25.
[9] Alistair EW Johnson, Tom J Pollard, Lu Shen, et al. 2016. MIMIC-III, a freely accessible critical care database. *Scientific data* 3 (2016), 160035.
[10] Denis Klimov, Alexander Shknevsky, and Yuval Shahar. 2015. Exploration of patterns predicting renal damage in patients with diabetes type II using a visual temporal analysis laboratory. *Journal of the American Medical Informatics Association* 22, 2 (2015), 275–289.
[11] Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics* 22, 1 (1951), 79–86.
[12] Anand Kumar, Daniel Roberts, Kenneth E Wood, et al. 2006. Duration of hypotension before initiation of effective antimicrobial therapy is the critical determinant of survival in human septic shock. *Critical care medicine* 34, 6 (2006), 1589–1596.
[13] Jianhua Lin. 1991. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information theory* 37, 1 (1991), 145–151.
[14] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. 2003. A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*. ACM, 2–11.
[15] Frank J Massey Jr. 1951. The Kolmogorov-Smirnov test for goodness of fit. *Journal of the American statistical Association* 46, 253 (1951), 68–78.
[16] Fabian Mörchen and Alfred Ultsch. 2005. Optimizing time series discretization for knowledge discovery. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM, 660–665.
[17] Robert Moskovitch and Yuval Shahar. 2015. Classification-driven temporal discretization of multivariate time series. *Data Mining and Knowledge Discovery* 29, 4 (2015), 871–913.
[18] Robert Moskovitch and Yuval Shahar. 2015. Classification of multivariate time series via temporal abstraction and time intervals mining. *Knowledge and Information Systems* 45, 1 (2015), 35–74.
[19] Robert Moskovitch and Yuval Shahar. 2015. Fast time intervals mining using the transitivity of temporal relations. *Knowledge and Information Systems* 42, 1 (2015), 21–48.
[20] Panagiotis Papapetrou, George Kollios, Stan Sclaroff, and Dimitrios Gunopulos. 2009. Mining frequent arrangements of temporal intervals. *Knowledge and Information Systems* 21, 2 (2009), 133.
[21] Dhaval Patel, Wynne Hsu, and Mong Li Lee. 2008. Mining relationships among interval-based events for classification. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. ACM, 393–404.
[22] Lucia Sacchi, Cristiana Larizza, Carlo Combi, and Riccardo Bellazzi. 2007. Data mining with Temporal Abstractions: learning rules from time series. *Data Mining and Knowledge Discovery* 15, 2 (2007), 217–247.
[23] Asaf Shabtai, Yuval Fledel, Yuval Elovici, and Yuval Shahar. 2010. Using the KBTA method for inferring computer and network security alerts from time-stamped, raw system metrics. *Journal in computer virology* 6, 3 (2010), 239–259.
[24] Asaf Shabtai, Uri Kanonov, and Yuval Elovici. 2010. Intrusion detection for mobile devices using the knowledge-based, temporal abstraction method. *Journal of Systems and Software* 83, 8 (2010), 1524–1537.
[25] Yuval Shahar. 1997. A framework for knowledge-based temporal abstraction. *Artificial intelligence* 90, 1-2 (1997), 79–133.
[26] Yuval Shahar. 1998. Dynamic temporal interpretation contexts for temporal abstraction. *Annals of Mathematics and Artificial Intelligence* 22, 1-2 (1998), 159.
[27] Yuval Shahar. 1999. Knowledge-based temporal interpolation. *Journal of Experimental & Theoretical Artificial Intelligence* 11, 1 (1999), 123–144.
[28] Yuval Shahar and Carlo Combi. 1998. Timing is everything. Time-oriented clinical information systems. *Western Journal of Medicine* 168, 2 (1998), 105.
[29] Dewang Shavdia. 2007. *Septic shock: Providing early warnings through multivariate logistic regression models*. Ph.D. Dissertation. Massachusetts Institute of Technology.
[30] Alexander Shknevsky, Yuval Shahar, and Robert Moskovitch. 2017. Consistent discovery of frequent interval-based temporal patterns in chronic patients data. *Journal of biomedical informatics* 75 (2017), 83–95.
[31] Mervyn Singer, Clifford S Deutschman, Christopher Warren Seymour, et al. 2016. The third international consensus definitions for sepsis and septic shock (sepsis-3). *Jama* 315, 8 (2016), 801–810.
[32] Fei Song and W Bruce Croft. 1999. A general language model for information retrieval. In *Proceedings of the eighth international conference on Information and knowledge management*. ACM, 316–321.
[33] Steven W Thiel, Jamie M Rosini, William Shannon, et al. 2010. Early prediction of septic shock in hospitalized patients. *Journal of hospital medicine: an official publication of the Society of Hospital Medicine* 5, 1 (2010), 19–25.
[34] CM Torio and BJ Moore. 2016. National inpatient hospital costs: the most expensive conditions by payer, 2013: statistical brief 204. (2016).