Continuously Predicting the Completion of a Time Intervals Related Pattern

Nevo Itzhak[1], Szymon Jaroszewicz[2,3], and Robert Moskovitch[1]

[1] Software and Information Systems Engineering,
Ben-Gurion University of the Negev, Beer Sheva, Israel.
nevoit@post.bgu.ac.il, robertmo@bgu.ac.il
[2] Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland.
[3] Faculty of Mathematics and Information Science,
Warsaw University of Technology, Warsaw, Poland.
szymon.jaroszewicz@ipipan.waw.pl

Abstract. In various domains, such as meteorology or patient data, events' durations are stored in a database, resulting in symbolic time interval (STI) data. Additionally, using temporal abstraction techniques, time point series can be transformed into STI data. Mining STI data for frequent time intervals-related patterns (TIRPs) was studied in recent decades. However, for the first time, we explore here how to continuously predict a TIRP's completion, which can be potentially applied with patterns that end with an event of interest, such as a medical complication, for its prediction. The main challenge in performing such a completion prediction occurs when the time intervals are coinciding, but not finished yet, which introduces an uncertainty in the evolving temporal relations, and thus on the TIRP's evolution process. In this study, we introduce a new structure to overcome this challenge and several continuous prediction models (CPMs). In the segmented CPM (SCPM), the completion probability depends only on the pattern's STIs' starting and ending points, while a machine learning-based CPM (CPML) incorporates the duration between the pattern's STIs' beginning and end times. Our experiment shows that overall, CPML based on an ANN performed better than the other CPMs, but CPML based on NB or RF provided the earliest predictions.

Keywords: continuous prediction · early prediction · temporal patterns

1 Introduction

Frequent temporal patterns, whether given by a domain expert or discovered by mining, were used already for temporal knowledge discovery, clustering, or classification [5,7]. Being able to continuously estimate, in real-time, whether a temporal pattern would fully occur, while its components are being revealed, is desirable, and can be useful in various applications, such as event prediction. Estimating the probability of the last pattern's component (e.g., an event of interest) occurrence, is of great interest. For example, predicting the death of a patient in the intensive care unit (ICU), based on continuous data, consisting of a temporal pattern that was observed in the data ending with death.

In many real-life data science problems, in which data are gathered from various sources, the multivariate temporal data are heterogeneous. Some variables may be sampled regularly but at different frequencies (e.g. sensor measurements) and some variables irregularly (e.g. variables measured manually or event-driven). Other temporal variables may be represented by events that may or may not have varying duration. In this study, we propose to employ the use of *temporal abstraction* [6,10] to transform the entire heterogeneous multivariate temporal data into meaningful symbolic time interval series. A *symbolic time interval (STI)* is a triplet of a start time, end time, and a symbol from an ordered alphabet.

Relation		Schematic Representation	Endpoints Representation
A before B	<	A B	A+ < A- < B+ < B-
A meets B	*m*	A B	A+ < A- = B+ < B-
A overlaps B	*o*	A B	A+ < B+ < A- < B-
A finished-by B	*fi*	A B	A+ < B+ < A- = B-
A contains B	*c*	A B	A+ < B+ < B- < A-
A starts B	*s*	A B	A+ = B+ < A- < B-
A equals B	=	A B	A+ = B+ < A- = B-

Fig. 1. Allen's seven temporal relations between a pair of STIs.

From STI data, frequent time intervals-related patterns can be discovered [10,4], which were shown in the past to be useful for knowledge discovery and as features for classification and prediction [5,11,7]. A *time intervals-related pattern (TIRP)* is comprised of a series of STIs and a set that defines all Allen's temporal relations (Fig. 1) between each of the pairs of STIs. For example, a pattern from time intervals data may be that hospitalized patients with COVID-19 frequently start with symptoms of "fever" and "cough," and a week later also begin experiencing shortness of breath, in which case the symptoms have not ended at the ICU admission time. Note that a TIRP's definition does not include the STIs' durations and their durations can vary among different instances. Using a frequent TIRP that ends with an event of interest, such as a patient's death, may allow for real-time continuous prediction of the completion of the TIRP's instance and of the occurrence of the event of interest. For example, the TIRP illustrated in Fig. 2 is defined by three STIs and three temporal relations, where the last STI, C, is considered as the event of interest.

Predicting a TIRP's completion is challenging due to the TIRP's instances variability which is reflected by the varying duration of the STIs, as noted earlier, and the varying duration of the gaps between the instances' STIs (e.g., in Fig. 2, between STI B and C). We define for the first time, as far as we know, the problem of predicting continuously the completion of a TIRP, and introduce novel models for the TIRP's continuous completion prediction.
The contributions of the paper are the following:

1. Defining the problem of continuous prediction of a TIRP's completion.
2. Introducing two novel methods for continuous prediction of a TIRP's completion.
3. A rigorous evaluation on real-life datasets, including new metrics to evaluate the continuous prediction of a TIRP's completion model.

2 Background

One of the forms of temporal abstraction is *state abstraction*, in which based on given cutoffs the time point values are categorized into symbols, and when adjacent time points have the same symbol, they are concatenated into a symbolic time interval. Several methods were proposed in the literature to learn the cutoffs from the data, such as equal width discretization (EWD), symbolic aggregate approximation (SAX) [6], and more [10].

A *symbolic time interval (STI)* $I = (s, e, sym)$, is a triplet of start time $s \in \mathbb{R}_{\geq 0}$, end time $e \in \mathbb{R}_{\geq 0}$, $e \geq s$ and a symbol sym ($sym \in \Sigma$) from an ordered alphabet Σ. A *time intervals-related pattern (TIRP)* Q is defined as a pair $Q = (IS, R)$, where $IS = \{I^1, ..., I^k\}$ is a series of k STIs and $R = \{r(I^i, I^j) : 1 \leq i < j \leq k\}$ is a set that defines all Allen's temporal relations (Fig. 1) between each of the $(k^2 - k)/2$ pairs of STIs in IS.

Given STI series data, TIRPs can be discovered for which several TIRP mining methods were proposed in the past two decades [10,4,11], most of which use Allen's temporal relations [1] that include seven temporal relations between a pair of STIs, as shown in Fig. 1. A TIRP is called frequent if its vertical support exceeds a predefined minimum threshold. Given a database DB of $|DB|$ unique entities (e.g., patients), the *vertical support* $VS(DB, Q)$ of a TIRP Q is defined as the cardinality of the set DB^Q of distinct entities within which Q holds at least once, divided by $|DB|$ (the total number of entities in DB), $VS(DB, Q) = |DB^Q|/|DB|$.

Frequent TIRPs are typically used as features for temporal data classification or prediction [5], as proposed first in [11]. Liu et al. [7] suggested a TIRPs-semantic-based probabilistic framework for STI data that can be used to answer varied semantic-level queries in a unified way, such as predicting future activities given observed ones. To the best of our knowledge, no previous study has investigated the task of continuous prediction of a TIRP's completion.

3 Methods

A model M predicts a TIRP Q's completion, given a database DB, by estimating the probability of observing the remaining part of Q, given its observed part at time t_c. An estimation is provided at each *current* time point t_c, and changes as a given Q's instance evolves over time. The database DB comprises $|DB|$ entities (e.g., patients), where each entity contains a series of STIs. We assume that in a specific STI series, STIs with the same symbols can not overlap.

Let p_{t_c} denote a prefix representing the observed part of Q at t_c, and s_{t_c} denote a suffix representing the remaining part of Q at t_c that is expected to occur. Thus, to estimate the Q's completion probability, at time point t_c, $Pr(Q \mid t_c)$, the following simple model can be used, which typically represents the confidence of a rule in sequential patterns:

$$Pr(Q \mid t_c) = Pr(s_{t_c} \mid p_{t_c}) = \frac{Pr(p_{t_c}, s_{t_c})}{Pr(p_{t_c})} = \frac{Pr(Q)}{Pr(p_{t_c})}. \quad (1)$$

The calculation in Formula 1 answers the question: "Out of all the times we saw p_{t_c}, how many times was it followed by s_{t_c} (i.e., Q has unfolded to completion)?" Thus, the number of times each p_{t_c} of Q occurs in the database and the number of times p_{t_c} is followed by s_{t_c} should be counted. Since the database DB comprises multiple entities, and each entity contains a lexicographically ordered STI series, instances of Q and p_{t_c} may be discovered more than once in a single entity. Each such instance is *counted separately* in the computation.

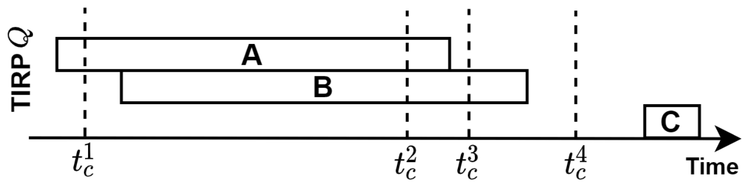

Fig. 2. TIRP Q's completion probability is estimated at any time point (e.g., t_c^2).

Applying Formula 1 to a relatively simple example sheds light on the challenges that arise while continuously predicting TIRP's completion. In Fig. 2, a TIRP $Q = \{A$ *overlaps* B, A *before* C, B *before* $C\}$ is shown, together with four time points t_c^1, t_c^2, t_c^3, and t_c^4 chosen to illustrate partial instances of the pattern to demonstrate various types of challenges. Calculating the numerator *Pr(Q)* is quite straightforward and is done by counting the number of times that A *overlaps* B is followed by A *before* C and B *before* C (i.e., Q). However, calculating the denominator $Pr(p_{t_c})$ is more challenging in some cases.

At t_c^4, the denominator $Pr(p_{t_c^4})$ is equal to the probability of seeing A *overlaps* B, which results in no uncertainty. Similar computations can be carried out at time points t_c^1 and t_c^3, but the situation is more complex since the time points are located after a starting point and before an ending point of an STI. Thus since an STI that is not finished yet is involved, p_{t_c} and s_{t_c} cannot be described with Allen's temporal relations. Instead, we need to use a different representation based on STIs' *tieps* (Def. 1).

Definition 1. (tiep) A *time interval endpoint* is a triplet $(t, type, sym)$ consisting of a time stamp $t \in \mathbb{R}_{\geq 0}$, an endpoint type, which can be either starting (+) or ending (-), and a symbol ($sym \in \Sigma$) from an ordered alphabet Σ.

Example 1. In Fig. 1, for an STI $A = (A_s, A_e,$ *"A"*), the starting and ending *tieps* are defined respectively as A+ $= (A_s,$ +, *"A"*) and A- $= (A_e,$ -, *"A"*).

A total order on *tieps* (Def. 1) is defined based on their time stamps, which are real numbers. Thus, the *tieps* can be used in inequalities defining temporal relations, while their structure will be exploited in the following sections.

At t_c^1, the $p_{t_c^1}$ is "A that has started but not ended yet," and thus, $Pr(A+)$ denotes the probability of seeing STI A that has started in DB. In the learning stage, in the database DB, each STI has its starting and ending *tieps*, and thus, the probability $Pr(A+)$ equals the probability of seeing STI A in DB. Similarly, at t_c^3, the $p_{t_c^3}$ can be represented by the following inequality between the STIs' *tieps*: $A+ < B+ < A-$.

However, in the learning stage, since STI B has started but not ended yet, its ending *tiep* B- has to satisfy $t_c^3 < B$- in database DB. Thus, the prefix's *tiep* ordering should be extended to $p_{t_c^3} = A+ < B+ < A- < B-$ to represent that STI B ended after B ended. The extended $p_{t_c^3}$ is equivalent to Allen's temporal relation A *overlaps* B (see Fig. 1). Uncertainty occurs at time point t_c^2, since $p_{t_c^2}$ includes STIs A and B that have already started but not yet ended (i.e., $t_c^2 < A-$ and $t_c^2 < B-$), it results in uncertainty about which temporal relation between A and B will finally unfold. Three different temporal relations are possible: *overlaps*, *contains*, or *finished-by*, which should be considered and used in Formula 1.

3.1 The Unfinished Coinciding STIs Challenge

Definition 2. (Unfinished STI) An *unfinished STI* I^* at time t_c is an STI whose starting *tiep* I^*+ satisfies $0 \leq I^*+ \leq t_c$ and whose ending *tiep* I^*- satisfies $t_c < I^*-$.

Throughout the text, the asterisk (*) will indicate that an STI is unfinished. The start time of an unfinished STI is known at time t_c, but its end time is not. In fact, it is censored: we only know that it is later than t_c.

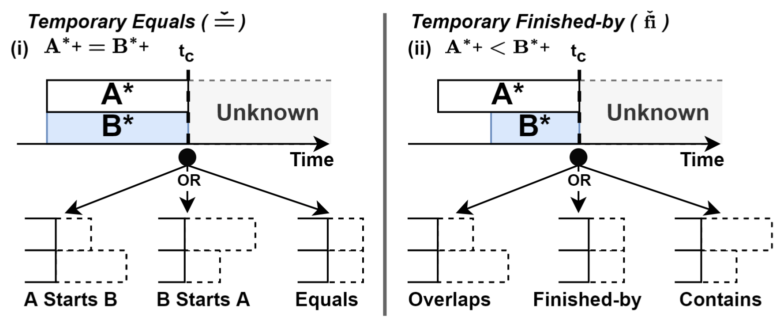

Fig. 3. The evolving temporary temporal relations.

A pair of unfinished coinciding STIs A^* and B^* may evolve into three possible temporal relations. The logic follows from the *tieps* representation of Allen's temporal relations that is presented in Fig. 1. Fig. 3.i shows that in the case of the *temporary equals* ($\check{=}$) temporal relation, their temporal relation may eventually evolve into A *starts* B, or B *starts* A, or stay at A equals B. The reason that A *starts* B and B *starts* A cannot be distinguished at t_c is that the exact temporal relation is determined by their end times, which are not yet known. Similarly, the *temporary finished-by* ($\check{\text{fi}}$) temporal relation shown in Fig. 3.ii, may eventually evolve into three possible Allen's temporal relations: A *overlaps* B, A *contains* B, or stay at A *finished-by* B.

3.2 TIRP-Prefixes

A TIRP can be represented by a series of starting and ending *tieps* instead of a series of STIs.

To maintain the conjunction of pairwise temporal relations among the STIs, the set of *tieps* has to be transformed into a sorted tieps' series, based on Allen's *tieps* representation (Fig. 1, right column).

A TIRP is divided into TIRP-prefixes (Def. 3) that are part of the TIRP's evolution process, which are created based on sub-sequences of the TIRP's *tieps*. In each TIRP-prefix, since the temporal relation between two unfinished STIs is uncertain, the temporary temporal relation $\check{r}$ is used to express the disjunction of possible final temporal relations based on the unfinished coinciding STIs challenge logic explained in Fig. 3.

Definition 3. (TIRP-prefix) Let Q be a TIRP of length k. A *TIRP-prefix* $\check{Q}$ of Q is defined as a pair $\check{Q} = (\check{IS}, \check{R})$, where $\check{IS}$ is a lexicographical ordered STI series of $\check{k} \leq k$ finished ($\check{IS}_f$) and unfinished ($\check{IS}_*$) STIs: $\check{IS} = \check{IS}_f \cup \check{IS}_*$, and $\check{R}$ is the set of all the temporal relations between each of the pairs of STIs in $\check{IS}$: $\check{R} = \check{R}_f \cup \check{R}_*$, where $\check{R}_f = \{r(I^i, I^j) : 1 \leq i < j \leq \check{k} \wedge \neg(I^i \in \check{IS}_* \wedge I^j \in \check{IS}_*)\}$ and $\check{R}_* = \{\check{r}(I^{*,i}, I^{*,j}) : 1 \leq i < j \leq \check{k} \wedge I^{*,i} \in \check{IS}_* \wedge I^{*,j} \in \check{IS}_*\}$.

Example 2. In Fig. 2, at t_c^2, it is known that $p_{t_c^2} = A^{*}+ < B^{*}+$, thus the TIRP-prefix is $\{A^* \ \check{\text{fi}}\ B^*\}$ and the three following TIRPs may potentially evolve into: $\{A$ *overlaps* $B\}$ or $\{A$ *finished-by* $B\}$ or $\{A$ *contains* $B\}$.

Algorithm 1 The TIRP-Prefix's Extender

Input: px - TIRP-prefix; **Output**: epx - extended TIRPs

1: unfPairs ← $px.\check{R}_*$; eRels ← ∅
2: **for each** pa **in** unfPairs **do**
3:     eRels ← eRels ∪ tempLogic(pa)
4: cmb = comb(eRels); epx ← ∅
5: **for each** c **in** cmb **do**
6:     cand ← cmb ∪ $px.\check{R}_f$
7:     **if** validTransitionTable(cand) **then**
8:         epx ← epx ∪ cand
9: **return** *epx*

In the TIRP-Prefix's Extender algorithm (Alg. 1), given a TIRP-prefix *px*, generates a set *epx* of all possible TIRPs that can evolve from *px*. The TIRP-prefix's temporary disjunctions of temporal relations (*unfPairs*, line 1) are set based on the rules presented in Fig. 3 by using the *tempLogic* function (lines 2-3). Then, the *comb* function generates a set of all the possible temporal relations that can be evolved given $\check{R}_*$, which is stored in *cmb*. Then, each *cmb* is joined with the TIRP-prefix non-temporary disjunctions of temporal relations $\check{R}_f$ and assigned to *cand*. Each *cand* represents a TIRP that can evolve from *px*.

However, *cand* can be a pattern with combinations of temporal relations that contradict each other. For example, A^* *overlaps* B^* and B^* *overlaps* C^*, the temporal relation between A^* and C^* can not be the relations *finished-by* or *contains*, but only *overlaps*. Thus, Allen's transition table [1] is used to reduce the number of generated candidates by avoiding impossible patterns (line 7).

Lastly, the potential evolved TIRPs *epx*'s instances have to be detected in the STI database DB by using rather a STIs based [10] or sequence-based [4] representation.

3.3 Continuous Prediction Models (CPMs)

In this section we will present the following two continuous TIRP completion prediction models:

SCPM The predicted TIRP's completion probability changes only at time points where *tieps* appear. As discussed in detail when Formula 1 was explained, the probability of TIRP Q completion is $Pr(Q)/Pr(p_{t_c})$, where p_{t_c} is a TIRP-prefix of Q at time t_c.

CPML The durations between consecutive TIRP-prefixes' *tieps* were used as features. For that, naive Bayes (NB) [9], random forest (RF) [2], and artificial neural network (ANN) [8] classifiers were used.

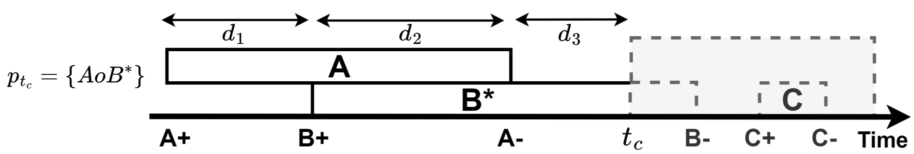

Fig. 4. Time durations d_1, d_2, and d_3 are based on *tieps* A+, B+, A−, and t_c, which are used as features for the classifiers to perform the TIRP's completion prediction.

Records for the classifier were created to represent the evolution of a TIRP over time. Multiple records were used as input for the classifier to include all the time stamps for each evolving TIRP-prefix instance. The TIRP's duration elements that were not observed until time point t_c are set to zero. Each instance's record target was set to whether the instance was finally unfolded into a TIRP's completion or not. For example, in Fig. 4, the TIRP-prefix instance $\{A$ *overlaps* $B^*\}$ is represented by the time durations between the four consecutive *tieps* ($f_1 = [A+,B+], f_2 = [B+,A-], f_3 = [A-,B-]$, and $f_4 = [B-,C+]$), which are used as features ($[f_1, f_2, f_3, f_4]$) for the classifier. For the instance at t_c, the record values are $f_1 = d_1$, $f_2 = d_2$, $f_3 = d_3$, and $f_4 = 0$.

The parameters for the models were selected after testing the performance of each considered combination. The parameters that performed best are the following: RF - maximum depth of 5, using bootstrap with 100 trees in the forest; ANN - two 50-neurons hidden layers, a maximum of 20 epochs, a batch size of 16, learning rate of 0.001 with gradually decreasing with early stopping, and with the ReLU activation function. NB, RF, and ANN were implemented with Python 3.6 and the Scikit-Learn package (`scikit-learn.org`) version 0.22.1. For parameters we did not specify, we used the package defaults.

3.4 Early Warning Strategies

Early warning strategies are used to decide that the TIRP will likely unfold once there is a high likelihood of the completion of the TIRP, based on the CPMs' estimated probabilities.

A decision that the TIRP will be unfolded is made immediately after the probability exceeds the prediction decision threshold (e.g., the gray point in Fig. 5) or when the threshold was consistently exceeded for some pre-defined time τ (e.g., the blue point in Fig. 5, which is defined with τ of three time stamps).

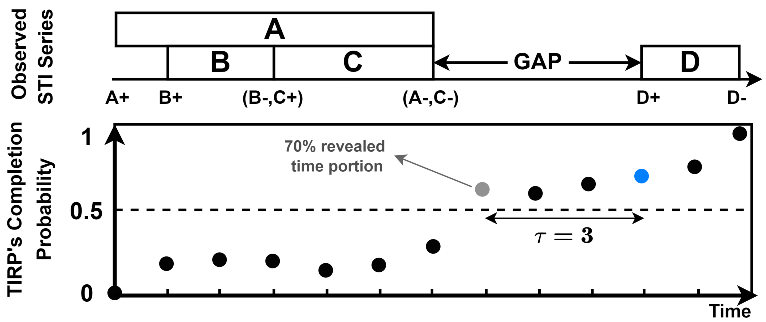

Fig. 5. The TIRP's completion probability at any time point is based on the observed STI series. The prediction decision is made when the completion probability is higher than the threshold, which is the horizontal dashed line (0.5 in this case), and a time delay τ has been passed.

4 Evaluation

Our goal was to evaluate the effectiveness of using continuous prediction models (CPMs) in predicting a TIRP's completion. The main *research questions* for this study were:

RQ1. Which CPM performs better, in terms of prediction performance and earliness, in predicting the completion of a TIRP?
RQ2. Which value of τ performs best, in terms of prediction performance and earliness, in predicting the completion of a TIRP?

4.1 Datasets

We evaluated the proposed models using real-life medical and non-medical datasets: cardiac surgical patients (CSP) [12], acute hypertensive episodes (AHE) [5], diabetes (DBT) [10], and elderly first injury fall (EFIF) [3] datasets. The events of interest were defined as the first occurrence of the following: CSP - cardiac index lower than 2.5 $L/min/m^2$, AHE - the target onset, DBT - HbA1C greater than 9%, and EFIF - first fall in elderly with a severe or moderate injury. Table 1 summarizes the main parameters of each dataset: entities (e.g., patients) number (#Ent), variables number (#Var), entities' maximum number of timestamps (#Timestamps), time granularity (Granularity), entities with the event of interest (#EntEvent), where the values in parentheses represent the percentage of #EntEvent out of #Ent, and the averaged different number of discovered TIRPs that ended with the events of interest (#TIRPs).

Table 1. The evaluation datasets' parameters

Name	#Ent	#Var	#Timestamps	Granularity	#EntEvent	#TIRPs
CSP	329	13	720	minutes	115 (35%)	257
AHE	1,000	4	238	hours	500 (50%)	246
DBT	1,710	12	24	months	239 (14%)	256
EFIF	823	15	144	weeks	121 (15%)	529

4.2 Experimental Setup

The models were evaluated on the ability to predict the completion of a TIRP that ended with an event of interest. Being able to predict continuously the completion of a TIRP, means it is possible to predict an event of interest. The entities' demographic data were not used, and only the time-based data were used for the continuous prediction. All the datasets were abstracted into STI series using SAX [6] with three symbols per variable. Then, TIRPs were discovered from the STI data using the KarmaLego algorithm [10] using Allen's seven temporal relations [1]. The patterns were discovered using 15% minimal vertical support from the entities that contained the event of interest.

Only the discovered patterns that ended with the event of interest were used. Yet, all entities, with or without the event of interest, were used to learn the model's parameters. TIRP-prefixes can be detected more than once in an entity's records, in which case each detected instance of the TIRP or its TIRP-prefixes were considered separately. The TIRP-prefixes instances from the training set were used to learn the model. To evaluate the models, all instances that started with the TIRP's earliest *tieps*, were used in the experiments. The events' STIs were considered instantaneous events, and the beginning of the event of interest was considered as the TIRP's completion.

Since each TIRP's completion was based on a different number of detected TIRP-prefixes' instances, the imbalance ratio differed between the patterns. We ran the experiments with ten-fold cross-validation, using target stratification. The instances of a TIRP and its TIRP-prefixes of the same entity appeared exclusively in the same fold.

Evaluation Metrics To evaluate the models' performance, a receiver operating characteristics (ROC) curve was calculated, together with the corresponding area under the curve (AUROC). The decisions were made based on a prediction decision threshold, that was varied between 0 to 1, and the decision time delay τ was varied using 0, 1, 2, and 3 time units. The ROC was created by varying the prediction decision thresholds between 0 to 1 (Fig. 5).

The *revealed time portion (RTP)* refers to the percentage of the instance that is revealed, at the time of the decision, relative to the entire instance's duration (start till its end time – known retrospectively). The revealed time is in percentage, since each instance, even of the same TIRP, may have a different duration. For example, in Fig. 5 the grey dot's RTP is 70%, where *D+* is considered as the event of interest and thus the instance's end.

4.3 Experiments and Results

Due to the limited space, each experiment is described first, followed by its results. The results are based on a total of 1,288 different TIRPs (Table 1). Each point on the charts represents the mean performance of the different TIRPs in each dataset, including confidence intervals of 95%.

Preliminary Analysis In this experiment, we present the performance in retrospect, analyzing the decision accuracy if the decision was made according to any of the RTPs (rather than according to the model threshold decision). Additionally, we wanted to understand whether there is an ideal RTP for a decision, or the more it reveals is better. The results were computed for all TIRP-prefixes instances for each RTP, based on the completion probability at this point.

Fig. 6 presents the mean AUROC at various TIRP-prefixes instances' RTP. The charts show that the more the instance is revealed, the more accurate predictions are provided by the models. RF and ANN perform best on all the datasets, except the EFIF dataset, in which the SCPM performs best above 40% revealed portion, while the CPMs perform worse. Thus, instance unfoldment can be predicted better when more information is revealed, but there is no optimal stage. However, early predictions are also desirable, so there is a trade-off, as we demonstrate in Fig. 8.

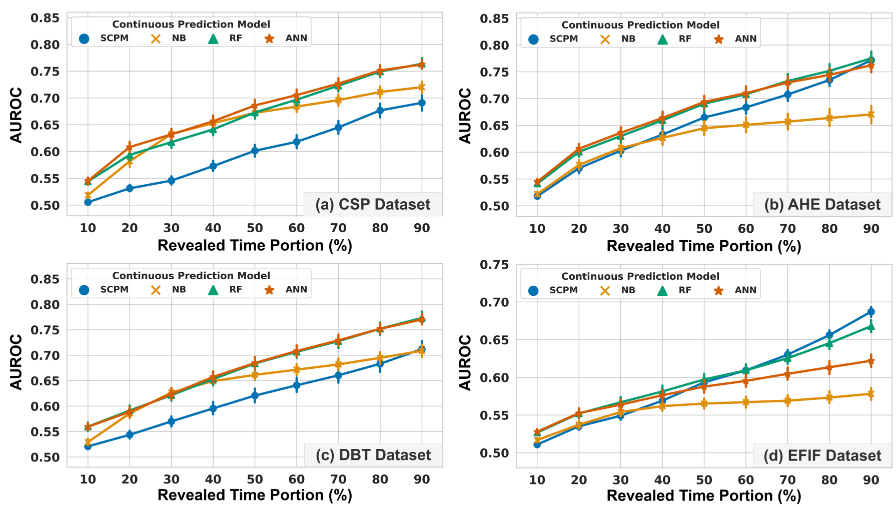

Fig. 6. The more the instance is revealed, the more accurate the predictions are. Overall, RF and ANN perform best on all the datasets, except the EFIF dataset.

Continuous TIRP's Completion Prediction In this experiment, we evaluated the models' ability to estimate the TIRPs' instances' completion, where the results were computed for all TIRP-prefixes' instances based on the decisions made by using the early warning strategies.

To answer RQ1, Fig. 7 presents the mean results of the models in predicting the TIRPs' instances completion with different values of τ. The ANN performed significantly best, except on the EFIF, in which SCPM performed best.

This implies that the duration distributions between TIRP-prefixes' consecutive *tieps* are similar between instances that ended with and without the event of interest on the EFIF dataset, and this may explain why SCPM performed better. The NB performed worst in all datasets.

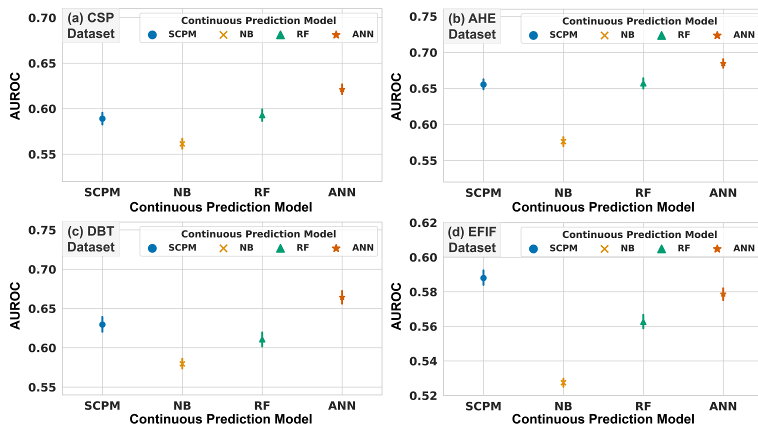

Fig. 7. ANN performed better than the other models with an average of 1.5% AUROC.

To answer RQ1 and RQ2, Fig. 8 presents the AUROC versus the mean instances' RTP of the corresponding decisions, for cases when the TIRP's completion was correctly predicted (true positive cases), for different models and values of τ, which are represented by five different colors and four different sizes, respectively. Overall, $\tau = 2$ performed best. While NB provided the earliest predictions for CSP, AHE, and EFIF, its prediction performances were poor. Also, for CSP, and EFIF, the SCPM, and ANN provided the latest but most accurate predictions. Except for the DBT dataset, there is a trade-off between prediction performance and earliness, where more accurate models need more time to make decisions. It strengthens the preliminary analysis, which showed the models were more accurate as time passed for each instance (Fig. 6).

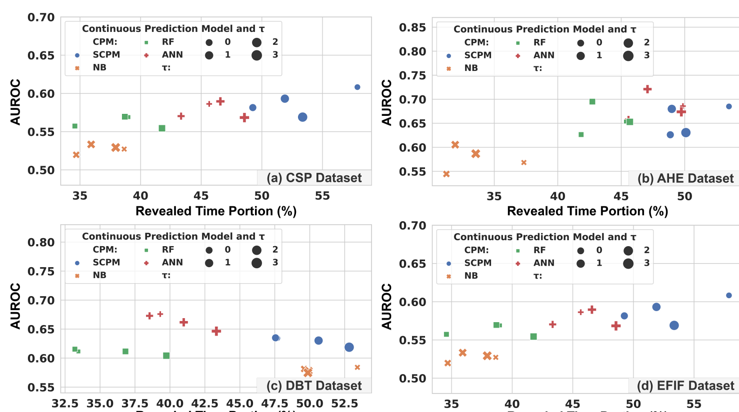

Fig. 8. More accurate models need more time to make decisions. Overall, SCPM provided the latest predictions, and NB and RF provided the earliest predictions.

5 Discussion

In this work, the continuous prediction of a TIRP's completion was studied for the first time. This approach can be useful with STI series databases, but what makes it more important and impactful is its use for heterogeneous multivariate longitudinal data, after employing temporal abstraction and transforming the data into STI series. Thus, it can be applied to any type of temporal variable, while incorporating any of them. The challenges, including the uncertainty of the evolving temporal relations, were discussed, and the TIRP-prefix representation and the extender algorithm (Alg. 1) to overcome this challenge were described. Based on that, the SCPM and CPML were proposed and a rigorous evaluation was performed on four real-life datasets. Overall, the CPML based on an ANN performed better than the other models with an average of 1.5% AUROC, but CPML based on NB or RF provided the earliest predictions. For future work, we intend to gear this methodology for event prediction, by applying it with multiple instances of various types of TIRPs that end with the event of interest.

Acknowledgments. Nevo Itzhak was funded by the Israeli Ministry of Science and Technology Jabotinsky scholarship grant #3-16643.

References

1. Allen, J.F.: Maintaining knowledge about temporal intervals. Communications of the ACM **26**(11), 832–843 (1983)
2. Breiman, L.: Random forests. Machine Learning **45**(1), 5–32 (2001)
3. Dvir, O., Wolfson, P., Lovat, L., Moskovitch, R.: Falls prediction in care homes using mobile app data collection. In: International Conference on Artificial Intelligence in Medicine. pp. 403–413. Springer (2020)
4. Harel, O., Moskovitch, R.: Complete closed time intervals-related patterns mining. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 4098–4105 (2021)
5. Itzhak, N., Pessach, I.M., Moskovitch, R.: Prediction of acute hypertensive episodes in critically ill patients. Artificial Intelligence in Medicine (2023)
6. Lin, J., Keogh, E., Wei, L., Lonardi, S.: Experiencing sax: a novel symbolic representation of time series. Data Mining and Knowledge Discovery **15**(2) (2007)
7. Liu, L., Wang, S., Su, G., Hu, B., Peng, Y., Xiong, Q., Wen, J.: A framework of mining semantic-based probabilistic event relations for complex activity recognition. Information Sciences **418**, 13–33 (2017)
8. McCulloch, W.S., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics **5**(4), 115–133 (1943)
9. Minsky, M.: Steps toward artificial intelligence. Proceedings of the IRE **49**(1), 8–30 (1961)
10. Moskovitch, R., Shahar, Y.: Fast time intervals mining using the transitivity of temporal relations. Knowledge and Information Systems **42**(1), 21–48 (2015)
11. Patel, D., Hsu, W., Lee, M.L.: Mining relationships among interval-based events for classification. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data. pp. 393–404. ACM (2008)
12. Verduijn, M., Sacchi, L., Peek, N., Bellazzi, R., de Jonge, E., de Mol, B.A.: Temporal abstraction for feature extraction: A comparative case study in prediction from intensive care monitoring data. Artificial Intelligence in Medicine **41**(1), 1–12 (2007)