This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3068895, IEEE Access

IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Learning to Rank for Multi-step Ahead Time-Series Forecasting

JIUDING DUAN[1], (Member, IEEE), HISASHI KASHIMA[2],
[1]Allianz Global Investors Japan, Co. Ltd., Minato-ku, Tokyo 106-0032 Japan (e-mail: jiuding.duan@gmail.com)
[2]Department of Intelligence Science and Technology, Kyoto University, Kyoto, 606-8501 Japan (e-mail: kashima@i.kyoto-u.ac.jp)

Corresponding author: Jiuding Duan (e-mail: jiuding.duan@gmail.com).

This work was partially supported by JSPS KAKENHI GrantNumber 20H04244.

ABSTRACT Time-series forecasting is a fundamental problem associated with a wide range of engineering, financial, and social applications. The challenge arises from the complexity due to the time-variant property of time series and the inevitable diminishing utility of predictive models. Therefore, it is generally difficult to accurately predict values, especially in a multi-step ahead setting. However, in domains such as financial time series forecasting, an ex-ante prediction of the relative order of values in the near future is sufficient; i.e., the next 100 days can help make meaningful investment decisions. In this paper, we propose a dynamic prediction framework that makes it possible to make an ex-ante forecast of time series with a special focus on the relative ordering of the forecast within a forward-looking time horizon. Through the lens of the concordance index (CI), we compare the proposed method with conventional regression-based time-series forecasting methods, discriminative learning methods and hybrid methods. Moreover, we discuss the use of the proposed framework for different types of time series and under a variety of conditions. Extensive experimental results on financial time series across a majority of liquid asset classes show that the proposed framework outperforms the benchmark methods significantly.

INDEX TERMS ranking estimation, time series, multi-step forecasting, concordance index, asset pricing, investment strategy.

I. INTRODUCTION

Time-series forecasting is a fundamental problem associated with a wide range of science, engineering, finance, and societal issues. In science and engineering applications, time-series forecasting is applied to areas such as energy management [1], predictive maintenance [2], and anomaly detection [3]. Normally, it is evaluated based on the now-casting performance, which reduces to certain evaluation metrics such as the tracking error. In the financial and social domains, the impact of time-series forecasting goes beyond nowcasting and it shifts its focus from the near future to the long-term horizon, bringing in other perspectives such as concordance, causality, in order to guide the decision makers to intervene appropriately. In this case, the use of a forecasted time series is prioritized over the conventional tracking error. Examples of this include empirical economics [4], [5], asset pricing [6], [7], business cycle analysis [8], monetary policy [9], and others that span all of the UN Sustainable Development goals, which address a blueprint for achieving a better and more sustainable future for all [10]. In addition to this discretion of appropriate utility function and evaluation metric, the length of the forward-looking horizon is an equally important aspect for such time-series forecasting task [11]. However, the majority of research work in time-series forecasting focuses on short-term forecasting, and often even on one-step ahead setting. For this reason, the potential variations in optimization objectives and evaluation metrics are not well explored beyond a predominant focus on tracking error. On the one hand, it is trivial to show the deterioration in forecast quality assuming a random walk prediction with no prior knowledge of what will happen in the next time epoch. On the other hand, the conviction that near-term nowcasting is accurate can provide meaningful support for long-term forecasting, especially in applications where the sequential dependency matters for multiple time epochs of interest. At the intersection of nowcasting and long-term forecasting, the predictability of time series is involved not only theoretically

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3068895, IEEE Access

but also empirically [12]. Assuming that the forecasting task is ergodic, the predictability can be formulated as the ratio of the variance in the optimal prediction to the variance in the ground truth time series. This sheds light on feasibility issues in time-series forecasting.

To address the many challenges in time-series forecasting, a variety of time-series forecasting approaches have been developed to capture certain structural assumptions of time series. Traditional methods include the non-stationary model [1], [13], the moving average method [14], the auto-regressive model [15], the auto-regressive moving average model [16], the auto-regressive integrated moving average (ARIMA) model [14], the tree-based model [17], and the fuzzy time series models that consider different types of uncertainties, varying from the formulation in evidence theory [18] to the formulation in statistics or fuzzy logic and set [19]. Independently, machine learning and deep neural network approaches, which have been developed in the past few decades with a focus on discriminating between observations, have also been adopted in time-series analysis to tackle the forecasting problem [17], [20]–[26]. Overall, the above techniques provide sound founding elements for time-series analysis and forecasting, but the simplicity of the model results in its limited capability to deal with sophisticated situations. Moreover, because many of these techniques are derived independently in terms of notations and terminology, the alignment and synergy between these methods become extremely challenging [27]. As a result, the implementation of machine learning methods in the context of time-series forecasting oversimplifies the time-series forecasting problem by assuming i.i.d. samples and neglecting the sequential nature of the observed signals. Although the actual utility of the forecast in a one-step ahead setting varies by application, the corresponding evaluation metric is often monotonously inherited from that of regression-based methods. In such scenarios, the common determinant criterion is the tracking error calculated from the point-wise difference between the ground truth and the forecasted value. Among the most widely used performance metrics in this category [28], the symmetric mean absolute percentage error (SMAPE) and the mean absolute scaled error are frequently used in existing literature [29], [30].

However, in a multi-step ahead setting, tracking error is no longer the only aspect of interest in performance evaluation [31]. Depending on how the forecasted values are further utilized, other discriminative metrics, such as directional symmetry [31], trajectory affinity [32], relative orders and concordance [33], become equally important or even more important when evaluating performance. For instance, in situations where privileged information is available, local forecasting around the privileged time epoch is less vulnerable from the perspective of tracking error [34], whereas the relative disordering phenomenon remains considerable. Among others, the relative ordering and concordance of the forecast are unique and critical to problems where the structural insights of time series matter [35].

Recently, the advancement of time-series forecasting methods has been featured by the construction of hybrid methods and the use of alternative perspectives. The traditional formulation and evaluation can be extended to a multi-step ahead forecasting setting by introducing a structured output, e.g., multiple output and recursive output. Such forecasting strategies and the essential techniques for multi-step ahead settings have been comprehensively reviewed by [36], [37]. Moreover, in contrast to traditional approaches, many studies have begun to adopt novel perspectives for time-series forecasting. Hybrid methods have proposed to highlight the benefits of combining the traditional time-series forecasting models with alternative objectives or complementary techniques [38], [39]. Some examples include the use of complex networks [30], [40], the ordered weighted averaging aggregation operator [29] and deep-neural-networks-based approaches [26], [34], [41]. These hybrid methods bring together different objectives from a mathematical programming perspective and some even enhance the expression power of existing architectures by incorporating deep neural networks [26], [42]. As with the growth of model complexity and the number of hyperparameters, the generalization of a model's predictive power in a multi-step forecasting setting becomes a greater challenge. Rather than contributing to this increasing sophistication, we argue that alternative objectives of certain time-series forecasting models, which are expected to deliver a higher explanation power by leveraging a limited number of model parameters, are critical to the time-series forecasting research. Specifically, in this study, we investigate the relative ordering objective, which has not yet been thoroughly formulated and explored.

In the financial domain, the relative ordering of the market values of asset prices at different time epochs is an essential component, because it is associated with theoretical and practical issues, e.g., mispricing, arbitraging, and market inefficiency. On the hunt for excessive investment returns, asset managers and hedge funds can leverage a variety of financial instruments, including futures, swaps, and options to monetize investment ideas and to optimize the investment performance given the investment ideas originated from such relative ordering. However, despite the wide adoption of stochastic mathematics and regression-based techniques in time-series analysis, limited efforts have been made to explore the relative orderings of time series within a specified horizon. Therefore, developing a method to forecast the relative ordering of the observations in time series is fundamental to time-series forecasting and critical in many financial applications [43], [44]. To an extreme degree, census data or variables in macroeconomics are released at a lower frequency in contrast to the data available at the exchange markets. This results in a gap between the tremendous number of model parameters and the small number of available observations [45], which further induces difficulty in model generalization. As an addition to the traditional econometric approaches, alternative methods

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3068895, IEEE Access

[46]–[48] have been proposed to achieve robust parameter estimation, where vector auto-regressive models [45], [48], [49] suffer from an out-of-sample prediction performance in the mid-long horizon. The intrinsic problem related to such application is that directional guidance and timing become more important issues than tracking error [50], [51]; therefore, alternative techniques that can leverage alternative utility functions of the forecast values should be highlighted [52].

In this paper, we propose a multi-step ahead forecasting framework that is capable of forecasting relative orders at multiple time epochs within a forward-looking horizon. The framework has three key components, including pairwise discriminative learning, local learning (LL) of privileged information, and dynamic multi-step ahead prediction with ex ante information. First, the pairwise discriminative learning module follows the learning-to-rank principles that directly optimise ranking objective rather than tracking error and output a ranking list instead of an array of forecasted values. Second, a LL algorithm is proposed to infer the values with the concept of neighboring or auxiliary samples, so that the optimised ranking list is interpretable and comparable to the conventional forecasts. Finally, all the above proposed elements are integrated into a dynamic multi-step ahead prediction scheme iteratively, aiming to boost the overall predictive performance. By bridging the best of two orthogonal evaluation metrics, i.e. relative ordering and tracking error, this scheme delivers an overlaying synergy of all the proposed elements underneath.

The layout of the paper is as follows. Section II describes the notations, the problem formulation, and some preliminaries. Section III introduces the proposed pairwise learning method for multi-step ahead time-series forecasting. Section IV presents the LL module and the overlaying dynamic prediction scheme. In Section VI, the dataset and the experimental settings are introduced. Section VII summarizes the key experimental results and discusses both the impact and the concerns regarding the proposed method. Section VIII draws conclusions and outlines proposals for future works.

II. PRELIMINARY

In this section, we describe the problem setting and the notations. As depicted in Section I, one major challenge in multi-step ahead forecasting is the domain-specific utility of the forecasted values. The other challenge is the diminishing utility of the trained model when the forecast horizon is expanded from one-step ahead to multi-step ahead.

To tackle the first challenge, we employ the learning-to-rank technique, which is considered as a fundamental method for ranking estimation, recommender systems, and data science [53]. Although being unexplored in time-series forecasting, the objective of learning-to-rank technique is to minimize the mismatch between the ground-truth ranking list and the estimated ranking list, which is in practice orthogonal to the conventional objective of minimizing the point-to-point euclidean error. In recent years, a variety of machine learning approaches to ranking estimation have been discovered via techniques such as Bayesian modeling [54], generalization of the Bradley-Terry model [55], the structured support vector machine (SVM) [56], optimal transport [57], the tree-based models [22], and fuzzy logic [29], [30], [40], etc. Different from the others, ranking SVM directly models and optimises the ranking loss on the data, which is essential to the ranking estimation problem. Although we only consider the linear ranking SVM in this study, the formulation is flexible for further extension to tackle the non-linear features via kernel tricks with a minor revision of the framework; therefore, ranking SVM is favourable among others, especially in applications where the availability of high quality features is limited. Overall, formulating time-series forecasting in a ranking setting remains an open and challenging problem. In this study, we leverage the ranking SVM because it tackles the ranking estimation problem with an end-to-end objective and with a favourable flexibility. In the following parts of this section, we establish the notations and define the problem setting, which are innovative in themselves.

To tackle the second challenge, we devise an iterative prediction scheme that follows the dynamic forecasting strategy in [36] in order to make informative inference. This scheme is motivated by LL techniques that bridge the two orthogonal worlds of tracking error and relative ordering. Although the details will be discussed in Section IV, the key notations and preliminaries will be described in this section.

A. PROBLEM SETTING

We consider multi-step time-series prediction. Assume $\mathbf{X}_{[0:N]}$ is the ground truth time series with a span of $[0:N]$. At time t, the observation of the time series is given as $X_{[t]} = (x_{[t]}, r_{[t]}) \in \mathbf{X}$, where $x_{[t]} \in \mathbb{R}^p$ is the observation of the time series and $r_{[t]} \in [0:N]$ is the associated rank of the observation within the whole span of the time series during $[0:N]$. We define $\mathcal{R} : \mathbb{R}^p \to [n]$ as an invertible ranking function that transforms $x_{[t]}$ to $\hat{r}_{[t]}$. The corresponding inverse function is denoted as $\mathcal{R}^{-1} : [n] \to \mathbb{R}^p$ and transforms $r_{[t]}$ to $\hat{x}_{[t]}$. The transform function is formulated as

$$\hat{r}_{[t]} = \mathcal{R}(\hat{x}_{[t]}), \quad (1)$$

$$\hat{x}_{[t]} = \mathcal{R}^{-1}(\hat{r}_{[t]}, x_{[0:t]}). \quad (2)$$

The goal is to make an informed forecast of a ranking list $\pi_{[t]}^{(h)} := \hat{r}_{[t+1:t+h]}$, which is a ranking list of length h that expresses the relative relationships of the observations that are obtained from the multi-step forecast $\hat{X}_{[t+1:t+h]} = (\hat{x}, \hat{r})_{[t+1:t+h]}$ at time $t = T$. Given all the historical observations $(X_{[0]}, X_{[1]}, ..., X_{[T]})$ at $t = T$, we aim to forecast the relative relationships in the time series in a forward-looking horizon of h during $[T+1 : T+h]$, where the ground truth of the forecast $r_{[T+1:T+h]}$ is a subset of $\mathbf{X}_{[0:N]}$. Without loss of generality, by setting

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3068895, IEEE Access

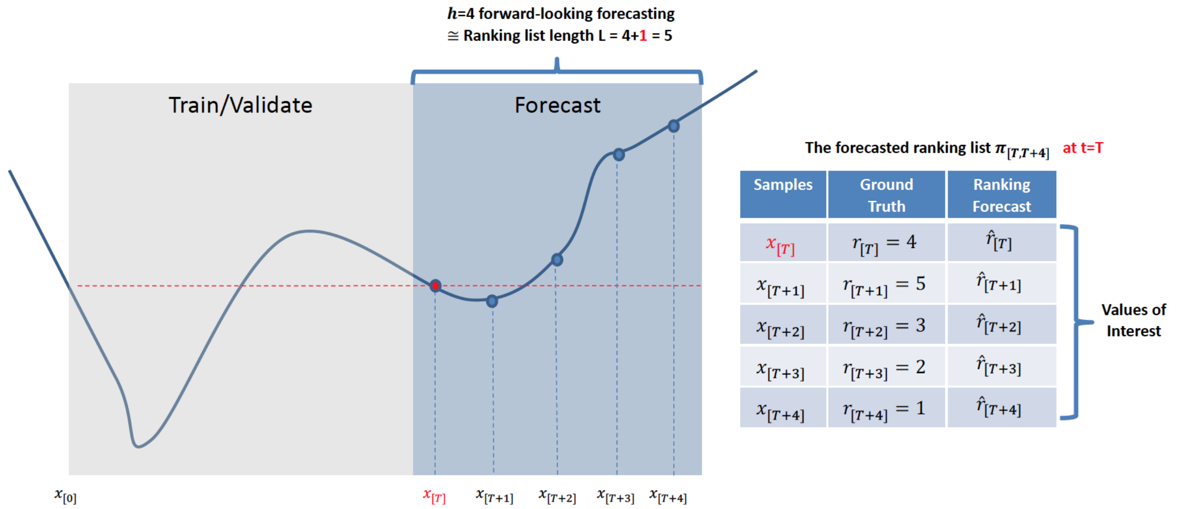


FIGURE 1. Illustration of the multi-step time-series forecasting scheme. There are in total five values of interest including the $\hat{r}_{[t]}$, whose ground truth of $x_{[t]}$ is known and highlighted in red color.

$p = 1$, we address the problem of forecasting time series with only one dimension. However, to implement all the competing methods and the proposed methods in a fair and qualitative manner, we construct multidimensional features based on generic feature engineering techniques for time series, especially time series in the financial domain.

In this study, we use $\pi^{(h)}_{[t]} \in [0 : h]$ to denote the h-step ahead forecast of relative relationships at time t. An auxiliary set $\tilde{x}_{[t-q:t]} = x_{[t-q:t]}$ representing the q auxiliary anchors is deployed in LL to boost the performance of the forecast. The simplest case is $q = 0$, indicating no privileged information other than the observation at time t can be utilized during the inference phase. However, in common cases, it is natural to assume that we have access to and can reuse all the historical observations up to time t. In the extreme scenario where $q = t$, the auxiliary set is identical to the training set and is denoted by $\tilde{x}_{[t-q:t]} = x_{[0:t]}$.

Figure 1 illustrates this relative-value-focused multi-step time-series forecasting scheme with an example that uses $h = 4$ and $q = 0$. In this extreme case, $x_{[t]}$ is the only available privileged information in inference, and therefore it becomes difficult to infer $\hat{x}_{[t+1:t+h]}$, regardless of whether the ranking estimation is perfectly aligned with the ground truth ranking or not within the test horizon. In a nutshell, only $x_{[t]}$ can be exploited for a hint as to whether the observations within the test horizon will be higher or lower than $x_{[t]}$. In cases where the relative ordering is not the only metric of interest, the learning-to-rank model requires deliberate revision before it can provide meaningful guidance.

B. FEATURE EXTRACTION

Among the various techniques that have been proposed to extract informative features from time series, we adopt two fundamental suites of indicators for financial time series, i.e., level indicators and momentum indicators [44].

Level indicators consist of the historical prices denoted by $F_{price}(t, i)$ and the moving averages of the historical prices [15] denoted by $F_{ma}(t, i)$, for $i \in \{5, 10, 20, 60, 120, 250, 500, 1000\}$, where

$$F_{price}(t, i; x_{[0:t]}) = x_{[t-i]} \tag{3}$$

$$F_{ma}(t, i; x_{[0:t]}) = \frac{\sum_{j=1}^{i} x_{[t-i]}}{i} \tag{4}$$

Momentum indicators include the moving average convergence divergence (MACD) [58] and rolling returns over the past $\{5, 10, 20, 60, 120, 250, 500, 1000\}$ days. In consideration of the fact that volatility is a critical facet of financial time series and the fact that the volatility scaling technique plays a significant role in the construction of investment strategies [59], we adopt the risk-adjusted momentum features $F_{rollingReturn}(t, i)$ by

$$F_{rollingReturn}(t, i; x_{[0:t]}) = \frac{x_{[t]} - x_{[t-i]}}{std(\dot{x}_{[t-i:t]})} \tag{5}$$

where $std(\dot{x}_{[t-i:t]})$ is the standard deviation of the daily price change $\dot{x}$ during the period $[t - i, t]$.

With the feature extraction defined as above, we construct multidimensional features for each observation $x_{[t]}$ that is obtained from the environment at time t. Finally, the level indicators and momentum indicators are combined by a concatenation denoted by

$$F_{all}(t, i; x_{[0:t]}) = [F_{price}, F_{ma}, F_{rollingReturn}] \tag{6}$$

where $i \in \{5, 10, 20, 60, 120, 250, 500, 1000\}$. Overall, 17 features are employed this study for the proposed framework.

Algorithm 1: Pairwise Ranking - Learning to rank for multi-step ahead time-series forecasting

Input: Forward-looking horizon h
Data: Observed time series $x_{[0:t]}$
initialization;
for $i \in [0:t]$ **do**
  $\mathbf{X}_{[i]} \leftarrow F_{all}(i; x_{[0:t]})$;
  $r_{[i]} \leftarrow \mathcal{R}(x_{[i]})$;
end for
$\hat{w}^* \leftarrow$ `Optimize Equation (15);`
for $j \in [0:h]$ **do**
  $\hat{\phi}(t+j) \leftarrow f(\mathbf{X}_{[t+j]}; \hat{w}^*)$;
  $\hat{r}_{[t+j]} \leftarrow \mathcal{R}(\hat{\phi}(t+j))$;
end for
Output: *Forecast of* $\hat{\pi}_{[t:t+h]}$

III. PAIRWISE LEARNING TO RANK

We formulate the multi-step time-series forecasting task as a multi-step learning-to-rank process that incorporates a learning phase and an inference phase. In the inference phase, a ranking model takes the historically observed objects and their associated feature matrices $\mathbf{X}_{[0:t]}$ as an input and outputs the inferred scoring of the observed objects $\hat{\pi}_{[0:t]}$.

$$\hat{\pi}_{[t:t+h]} = f(\mathbf{X}_{[0:t]}, \mathbf{W}; w) \tag{7}$$

The overall learning objective of the ranking model is to discriminate the observed objects from a relative relationship perspective so that the inferred ranking list $\hat{\pi}_{[t:t+h]}$ is close enough to the ground truth $\pi_{[t:t+h]}$. Assuming the observations being forecasted, $\mathbf{X}_{[0:t]}$, by the ranking model follow the same distribution of historical observations, the learning objective in Equation (7) is approached by devising a pairwise ranking model based on the observed time series by time t. Given a rank order $r_{[i]} \in \pi_{[0:t]}$ where $i \in [0:t]$, a scoring function $\phi(\mathbf{X}_{[i]})$ is devised to measure the relative ordering of the objectives.

$$\begin{aligned} \phi(i) &= \phi(\mathbf{X}_{[i]}; w) && (8)\\ &= \sum_j w^j \mathbf{X}^j_{[i]} && (9) \end{aligned}$$

An array of such scoring $\phi(\cdot)$ can be trivially converted to a corresponding ranking list by using Equation (1). In fact, it is identical to the primal model output $\hat{\pi}_{[t,t+h]}$ from the ranking perspective and is used as an input in the calculation of the ranking performance metric, which we will introduce later in Section V.

Assume that $\mathbf{W} \in \mathbb{R}^{t,t}$ is a match-up matrix indicating the soft constraints that can be incorporated into the learning part of the framework, where an entry $W_{i,j}$ denotes the importance weight of each pairwise comparison (i,j) in the ranking list. Given the ground truth $\pi_{[0,t]}$, each entry in the match-up matrix $W_{i,j}$ is defined as follows:

$$W_{i,j} = \begin{cases} +1 & \text{if } r_{[i]} \succeq r_{[j]} \\ -1 & \text{if } r_{[i]} \prec r_{[j]} \\ 0 & \text{pair } (i,j) \text{ is not considered} \end{cases}$$

where $i,j \in [0:t]$.

The ranking loss between the ground truth π and the inferred ranking list $\hat{\pi}$ is expressed as

$$\Delta(i,j) = W_{i,j}(\phi(\mathbf{X}_{[i]}; w) - \phi(\mathbf{X}_{[j]}; w)) \tag{10}$$

where $\phi(\mathbf{X}_{[i]}; w)$ is the scoring function defined by Equation (8) for the observation at time i.

Note that the feature for the observation at t is often a function of the observation by that time; however, the values are not available between t and $t+h$ in multi-step ahead setting. Therefore, the construction of pseudo features is necessary for forecasting the future observation at $t+h$. As a naive solution, we adopt the latest available observation for the ex-ante observations at each future time epoch $t+h$, formulated as

$$\tilde{x}_{[t+j]} = x_{[t]} \tag{11}$$

for all $j \in [0:h]$. Therefore, the features $\tilde{\mathbf{X}}_{[t+h]}$ are calculated by calling the feature construction function with the above pseudo estimates by

$$\tilde{\mathbf{X}}_{[t+h]} = F_{all}(t+h; x_{[0:t]}, \tilde{x}_{[t:t+h]}) \tag{12}$$

The overall loss function is written as

$$L(\mathbf{X}, \mathbf{W}; w) = \frac{1}{|\mathcal{P}||\mathcal{N}|} \sum_{x \in \mathcal{P}} \sum_{y \in \mathcal{N}} d(\Delta(i,j)) \tag{13}$$

where

$$d(\Delta(i,j)) = \max(0, 1 - \Delta(i,j)) \tag{14}$$

is the Hinge loss [60]. In Equation (13), $\mathcal{P}$ and $\mathcal{N}$ denote the upper-ranked object subset and the lower-ranked object subset, respectively. Without loss of generality, one can extend the definition of $\mathcal{P}$ and $\mathcal{N}$ to the full list, which is also referred to as the list-wise ranking operation [56].

In our proposed framework, we restrict ourselves to solving the classic RankSVM optimization problem [56], [60], [61]. The overall optimization objective is expressed as

$$\min_w \frac{\lambda}{2}||w||^2 + L(\mathbf{X}, \mathbf{W}; w) \tag{15}$$

where λ is the hyperparameter regularizing the learning objective in SVM via the soft margin embedded in the first term of the optimization objective in Equation (15).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3068895, IEEE Access

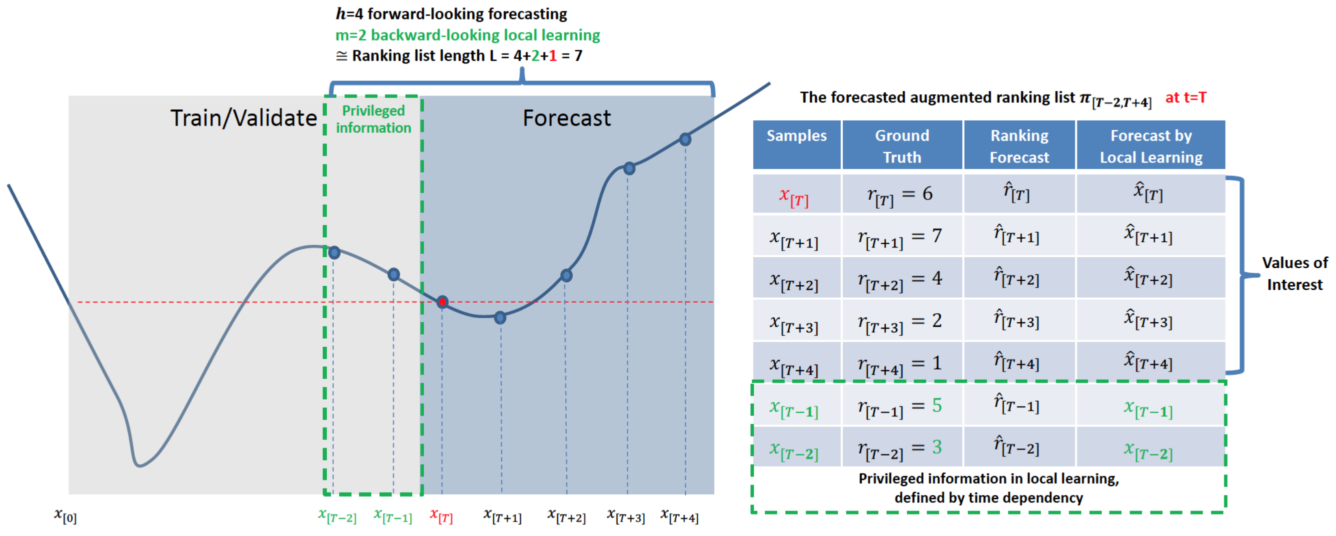


FIGURE 2. Illustration of the local learning that is embedded into the dynamic prediction scheme for multi-step time-series forecasting scheme

---

Algorithm 2: Local learning of $\hat{x}_{[t+1:t+h]}$

One-Step **Input:** Backward-looking horizon m
Data: Observed time series $x_{[0:t]}$
initialization;
$\hat{r}_{[t:t+h]} \leftarrow$ `Algorithm 1`;
$\hat{r}_{[t-m:t]} \leftarrow$ `Equation (7)`;
$\hat{r}_{[t-m:t+h]} \leftarrow$ `Merge` $\hat{r}_{[t-m:t]}, \hat{r}_{[t:t+h]}$;
for *each* $j \in [t+1:t+h]$ **do**
  $j^+ \leftarrow \underset{p\in[t-m:t]}{\operatorname{argmin}} \hat{r}_{[p]} - \hat{r}_{[j]}$ for $\hat{r}_{[p]} > \hat{r}_{[j]}$;
  $j^- \leftarrow \underset{p\in[t-m:t]}{\operatorname{argmax}} \hat{r}_{[p]} - \hat{r}_{[j]}$ for $\hat{r}_{[p]} < \hat{r}_{[j]}$;
  $\hat{x}_{[j]} = \frac{1}{2}(x_{[j+]} + x_{[j-]})$;
end for
Output: Forecast of $\hat{x}_{[t+1:t+h]}$

A. OPTIMIZATION

Solving for w^* in Equation (15) is a numerical optimization problem. Generic solutions to such a problem include the stochastic subgradient descent method in the Support Vector Machine optimization [62], [63]. Specifically, we adopt the stochastic subgradient descent method for solving the L2-regularized L1-loss SVM implemented in the scikit-learn library [64] as the solver for our proposed pairwise learning-to-rank method. The subgradient update is given as

$$w \leftarrow w - \eta \nabla_w (W_{i,j} w^T (\mathbf{X}_{[i]} - \mathbf{X}_{[j]})) \tag{16}$$

where $\eta > 0$ is the learning rate and is chosen as $\eta(t) = \frac{1}{\lambda(t+t_0)}$. t is the epoch of gradient update and t_0 is chosen by the heuristic proposed in [65]. For a cross-temporal pair (i, j) sampled from the training set, the corresponding subgradient is estimated by

$$\nabla_w(i,j) = \begin{cases} \lambda w & \text{if } W_{i,j}(\phi(i) - \phi(j)) > 1, \\ \lambda w - W_{i,j}(\mathbf{X}_{[i]} - \mathbf{X}_{[j]}) & \text{otherwise} \end{cases}$$

The selection of regularization term λ was done through a set of predefined hyperparameters $\{10^{-4}, 10^{-2}, ..., 10^4\}$. Empirically, we would like to emphasize that the pairwise ranking approach faces certain issues like scalability when the volume of training data grows and computational convergence when the learning rate is not chosen properly. However, with a fine-tuning of the key hyperparameters, e.g. the learning rate η and the λ associated with the regularization term as we suggested, these issues can be mitigated.

Algorithm 1 describes the proposed learning-to-rank framework for multi-step ahead time-series forecasting. The input of the algorithm is the forward-looking horizon. After initialization of the model parameters including w, λ, and the forward-looking horizon h, the pairwise learning-to-rank model is trained until the convergence condition is met. This training phase is followed by a standalone inference phase where the forecast of relative ordering $\hat{r}_{[t:t+h]}$ is inferred.

Note that the $\hat{r}_{[t:t+h]}$ estimated by Algorithm 1 is insufficient for an estimation of the tracking error, which is normally adopted by competing methods and related works on multi-step ahead time-series forecasting. To fill in this gap between the orthogonal evaluation metrics, we further develop a LL technique in Section IV that is able to obtain $\hat{x}_{[t+1:t+h]}$ and integrate the LL module into an overlaying dynamic forecasting scheme that operates iteratively between learning and inference to lower the tracking error while optimizing the discriminative objective in Equation (15).

IV. DYNAMIC MULTI-STEP FORECASTING

To improve the quality of the multi-step ahead forecast, we apply LL techniques during the approximate inference of $\hat{x}_{[t+1:t+h]}$. In Algorithm 1, the output of the algorithm is the relative ordering of objects, which reveals only the relative relationship between future observations and the observation at time t. We devise a local learning(LL) procedure to improve the inference of $\hat{x}_{[t+1:t+h]}$, which can be used in turn to improve the quality of features by letting

$$\tilde{x}_{[t+j]} = \hat{x}_{[t+j]} \tag{17}$$

where $j \in [1 : h]$ and $\tilde{x}_{[t]}$ represents the ex-ante expected value of the time series at time t.

Figure 2 illustrates the proposed LL scheme. In this study, we define the privileged information available for local learning from the perspective of time dependency [11]. At time T, the most recent $m \leq T$ samples are employed for local learning and we update the point estimate each time when the local learning procedure is called.

To further boost the forecasting performance based on the informative ex-ante prediction obtained from local learning, we devise a dynamic prediction as described in Algorithm 3.

A. LOCAL LEARNING

Local learning is an important technique in machine learning that allows the model to efficiently incorporate certain dependency structures, such as neighborhood dependency [66], time dependency [11], and spatial-temporal dependency [42]. Given an output $\hat{r}_{[t:t+h]}$ from Algorithm 1, auxiliary information can be added as anchors into the well-trained ranking model, so that the inference of $\hat{x}_{[t+1:t+h]}$ can be formulated as a LL procedure as described in Algorithm 2.

Assuming m neighbors are available for the LL, we rewrite Equation (1) in array form and incorporate the local neighbors as anchors by

$$\hat{x}'_{[t+1:t+h]} = \tilde{\mathcal{R}}^{-1}(\hat{r}_{[t-m:t+h]}, x_{[t-m:t]}) \tag{18}$$

The effectiveness of such an ex-ante forecast can be verified via performance metrics such as the concordance index (CI). Despite of the orthogonality between CI and tracking error, this promising result sheds light on a potential improvement in terms of the ex-ante tracking error; therefore, we propose an iterative dynamic forecasting scheme to capture this signal and combine the best of LL framework with the proposed learning-to-rank framework for time-series forecasting. In Figure 2, we illustrate a toy example of local learning, where we indent to forecast five values of interest in the forward-looking horizon, with two privileged samples in the training and validation set highlighted in green color. In this example, the total length of the ranking list is seven, among which five are identical to the values of interest illustrated in Figure 1. What makes a difference is that we have partial access to their ground truth; therefore, a performance improvement in the inference phase can be expected.

Algorithm 3: DynaPairwise Ranking - Dynamic prediction scheme that improves learning to rank for time-series forecasting and local learning iteratively

Input: Maximum number of iteration l_{max}
Data: Observed time series $x_{[0:t]}$
initialization;
$\hat{r}_{[t:t+h]} \leftarrow$ `Algorithm 1;`
while $l \leq l_{max}$ **do**
  $\hat{x}_{[t+1:t+h]} \leftarrow$ `Algorithm 2;`
  $\hat{r}_{[t:t+h]} \leftarrow$ `Algorithm 1;`
  l++;
end while
Output: Forecast of $\hat{x}_{[t+1:t+h]}$, $\hat{r}_{[t+1:t+h]}$

B. DYNAMIC PREDICTION

The previous section showed that multi-step ahead forecasting can be cast in a conventional supervised learning-to-rank framework by employing certain inference techniques such as LL. In this section, we extend the proposed framework in Algorithm 2 and further propose an iterative dynamic prediction scheme that improves the performance of multi-step ahead forecasting. The proposed scheme follows the recursive strategy for multi-step ahead time-series forecasting [11]. Such recursive strategies for time dependency are widely deployed implicitly or explicitly in structured output prediction models such as conditional random fields [67] and recursive neural networks [68].

Overall, the proposed iterative dynamic scheme involves three steps as described in Algorithm 3. The key components are organized in a nested structure. In the first step, Algorithm 1 is called to give an initial ex-ante forecast of the relative ordering $\hat{r}_{[t+1:t+h]}$. Next, the LL procedure depicted in Algorithm 2 is called to provide an initial ex-ante forecast of the time series $\hat{x}'_{[t+1:t+h]}$. Because the features are constructed by a function of such an ex-ante estimate of the time series in Equation (6), Algorithm 1 is called again in step 3 to brush up the features by using these ex-ante estimations. Such an iterative procedure can be terminated either when a convergence criterion is met or when the maximum number of iterations l_{max} is reached. In practice, we let $l_{max} = 1$ and apply the dynamic prediction procedure on a rolling basis with a specified interval that equals to the forward-looking horizon h.

V. EVALUATION METRICS

In line with the previous studies, three evaluation metrics, i.e., SMAPE [69], RMSE [28], and the CI [70], [71] are used to evaluate the performance of competing models including the theta model [72], [73], the regression-based model [74], the ARIMA model [14], the DeepAR model

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3068895, IEEE Access

TABLE 1. Hyperparameters deployed in the benchmark scheme and the proposed method.

Parameter	Setting	Description
N_{train}	500	size of the training and validation data
N_{roll}	{50,100,200}	number of time epochs between model retraining
m	{0,100,500}	length of the backward-looking horizon in local learning
h	{50,100,200}	length of the forward-looking forecast horizon
$l_{\max}$	1	number of iterations in the dynamic prediction scheme
λ	$\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$	strength of the regularization term that controls the soft margin
μ	refer to Table 2	calibration factor that reflects domain knowledge
η	refer to Equation (16)	learning rate of the gradient descent method

[26], the random forest model [17], the LightGBM model [22], the logistic regression model [75], [76], and the ordinal regression method [77].

Mathematically, the metrics are calculated as follows:

$$\text{SMAPE} = \frac{1}{n}\sum_{t=1}^{n}\frac{\left|\hat{x}_{[t]} - x_{[t]}\right|}{\left|\hat{x}_{[t]}\right| + \left|x_{[t]}\right|} \times 100\% \quad (19)$$

$$\text{RMSE} = \frac{1}{n}\sqrt{\sum_{t=1}^{n}\left|\hat{x}_{[t]} - x_{[t]}\right|^2} \quad (20)$$

$$\text{CI} = \frac{2}{|\mathcal{E}|}\sum_{(i,j)\in\mathcal{E}} \mathbb{1}(\hat{r}_{[i]} \succeq \hat{r}_{[j]}) \quad (21)$$

where $\mathcal{E} := \{(i,j); x_{[i]} \geq x_{[j]}, i < j \in [1:n]\}$ is the set of event of interest (EOI), which is observed in the ground truth time series. $\mathbb{1}(\hat{r}_{[t]} \geq \hat{r}_{[t]})$ is an indicator function that outputs 1 when the condition $\hat{r}_{[t]} \geq \hat{r}_{[t]}$ is satisfied. At time i, we focus on the concordance between the relative ordering of the observations and the relative ordering of the model forecast. The CI is a generalization of the area under the ROC curve to the regression problem, and therefore it can be calculated trivially by transforming the continuous outputs into a ranking list using Equation (1). In our definition, over a specific time horizon n, the CI is 1 if all the upward movements of the time series along the timeline are successfully captured by the forecast and 0 if all the upward movements of the time series are forecasted as downward movements. Assuming the time series is a random walk with no expectations regarding its upside and downside, CI $= 0.5$ indicates that the performance is as good as a random predictor. However, given some domain knowledge of the time series, the outlook for certain directional move of the time series might shift from neutral to up or down in expectation. In such cases, the outputs of the models are calibrated toward the predefined expectation by

$$\hat{x}'_{[t_0]}(t) = \hat{x}_{[t_0]}e^{\mu(t-t_0)} \quad (22)$$

where μ is a positive scalar when the outlook for the time-series observation from time t_0 is more on the upside and is negative when when the outlook is more on the downside.

Note that for discriminative models, including logistic regression, ordinal regression, and the proposed pairwise ranking models, only $\hat{r}_{[t:t+h]}$ is forecasted. Therefore, only the CI is calculated and reported in the comparison with the competing methods. An exception is the proposed dynamic prediction scheme assembled with LL, where the output of the discriminative model can be interpreted as scalar values $\hat{x}_{[t:t+h]}$ given sufficient privileged information. For non-discriminative models, including the theta model and the regression-based models, both $\hat{r}_{[t:t+h]}$ and $\hat{x}_{[t:t+h]}$ are forecasted. All three evaluation metrics are calculated and reported in Section VI.

VI. EXPERIMENT

In this section, we describe the characteristics of the dataset and the benchmark procedure that is executed for all competing and proposed methods. Throughout the experimental results, particularly in Table 3 and Table 4, we refer to Pairwise Ranking and DynaPairwise Ranking as the proposed Algorithm 1 and the proposed Algorithm 3, respectively.

A. DESCRIPTION OF DATASETS

We use historical future contract and index data obtained from the Bloomberg Terminal[1]. The dataset is retrieved on October 31, 2020, and consists of major liquid asset classes including equity indices, fixed income indices, and currency indices[2]. An overview of the dataset is provided in Table 2. For the equity and fixed income indices, the observations are either price return or total return, which are tradable in the market via exchange-traded funds or future contracts at low transaction costs. The time horizon of the time-series data ranges from December 31, 1999, to December 31, 2019, on a daily basis. For currency indices, the time horizon is from December 31, 1989, to December 31, 2019, on a daily basis. In the last column of Table 2, the historical return statistics are presented to characterize the asset classes. In the last column of Table 2, we summarize the average annualized rolling return and the standard deviation of the annualized rolling return over the entire observation period. Because the observation period contains at least one economic and market cycle, the return and standard deviation statistics

[1] https://www.bloomberg.com/professional/

[2] Financial indices source data can also be obtained from other data vendors, e.g. https://tradingeconomics.com/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3068895, IEEE Access

TABLE 2. Overview of financial time series dataset. The dataset has two categories, i.e., growth indices and mean-reverting indices, addressing the economic properties of the underlying asset classes. In the upper part of the table, equity indices and fixed income indices are categorized as growth indices. In the lower part of the table, currency exchange indices are categorized as mean-reverting indices.

Index	Asset Class	Ticker	Description	Experiment Period	Avg. Rolling Return (std.)
Growth Indices					
S&P 500	Equity index	SPX	S&P 500 price return index	12.31.1999 to 12.31.2019	5.19%(16.11%)
Russel 2000	Equity index	RUT	Russell 2000 index	12.31.1999 to 12.31.2019	7.27%(19.21%)
DAX	Equity index	DAX	German stock total return index	12.31.1999 to 12.31.2019	5.25%(21.96%)
TOPIX	Equity index	TPX	Tokyo Stock Exchange price index	12.31.1999 to 12.31.2019	2.58%(22.34%)
Bund 10Y	Fixed income index	CBKIG0FT	German 10Y Bund futures index	12.31.1999 to 12.31.2019	4.60%(4.82%)
Gilt 10Y	Fixed income index	CBKIK0FT	UK 10Y Gilt futures index	12.31.1999 to 12.31.2019	3.60%(5.45%)
Treasury 10Y	Fixed income index	CBKIU0FT	US 10Y Treasury futures index	12.31.1999 to 12.31.2019	3.82%(5.13%)
Mean-reverting Indices					
EURUSD	Currency	EURUSDCR	EURUSD carry return index	12.31.1989 to 12.31.2019	0.29%(10.80%)
GBPUSD	Currency	GBPUSDCR	GBPUSD carry return index	12.31.1989 to 12.31.2019	1.58%(9.29%)
CHFUSD	Currency	CHFUSDCR	CHFUSD carry return index	12.31.1989 to 12.31.2019	1.13%(10.81%)
JPYUSD	Currency	JPYUSDCR	JPYUSD carry return index	12.31.1989 to 12.31.2019	−0.72%(11.05%)
CADUSD	Currency	CADUSDCR	CADUSD carry return index	12.31.1989 to 12.31.2019	0.49%(7.85%)
AUDUSD	Currency	AUDUSDCR	AUDUSD carry return index	12.31.1989 to 12.31.2019	2.89%(13.10%)
NZDUSD	Currency	NZDUSDCR	NZDUSD carry return index	12.31.1989 to 12.31.2019	4.30%(13.43%)
NOKUSD	Currency	NOKUSDCR	NOKUSD carry return index	12.31.1989 to 12.31.2019	1.44%(11.89%)
SEKUSD	Currency	SEKUSDCR	SEKUSD carry return index	12.31.1989 to 12.31.2019	0.79%(12.76%)

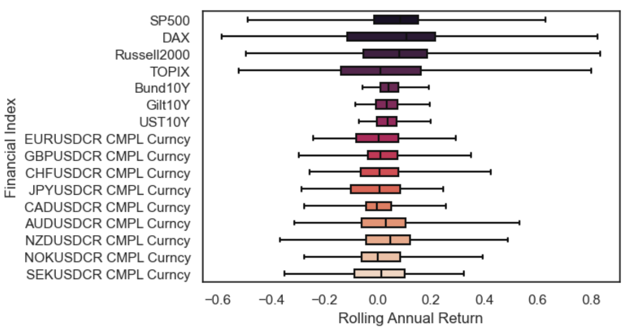


FIGURE 3. Categorizing the financial indices into growth indices (in blue) and mean-reverting indices (in red). Equity and fixed income indices are categorized as growth indices because of their positive expected return over the market cycle. Currency indices are categorized as mean-reverting indices because of their strong volatility with a mean at almost zero.

reflect the expected growth of an asset class in mid-long term. Depending on the level of growth expectation, we categorize the 16 indices into two groups. On the one hand, equity and fixed income indices are categorized as growth indices, because they generally record positive asset pricing results in the historical performance. This is in line with the economic outlook of these asset classes in the long run. On the other hand, currency indices are categorized as mean-reverting indices, because the annualized returns of the currency indices are not significantly positive in a market cycle, which signals the strong mean-reversion style of the asset class.

B. EXPERIMENTAL SETTING

All results of both competing and proposed methods are produced based on a common benchmark procedure. For each of the retrieved time series, we retrained on a rolling basis, i.e., every N_{roll} trading days and varied the forecasting horizon from 50 trading days to 200 trading days accordingly. By default, the forward-looking forecast horizon h is aligned with the interval of the rolling retraining by setting $h = N_{\text{roll}}$. By varying the horizon from short term $h = 50$ to mid-long term $h = \{100, 200\}$, we validate the consistency of the results and report the evaluation metrics together with their statistical significance, i.e. standard deviation. To summarize, the benchmark procedure is analogous to Algorithm 1, where the LL in Algorithm 2 and the iterative dynamic prediction in Algorithm 3 are not considered.

For each of the individual time series, we take the last 30% of the retrieved data as the test set and report the performance based on the forecasted values in that horizon. Given the benchmark scheme that behaves in a rolling manner, we set the latest $N_{\text{train}} = 500$ trading days as the input data for all the models involved and set $m = N_{\text{train}}$ for Algorithm 2 by default, so that all observed data are exploited in LL. In practice, the results delivered by setting $m = 100$, $m = 500$ and $m = 1,000$ does not show a significant difference; therefore, we report all results by setting $m = N_{\text{train}}$ as the default configuration. The detailed hyperparameters for the benchmark procedure and the proposed learning algorithm are summarized in Table 1.

For the competing benchmarks, we follow the standard de facto hyperparameter tuning processes and models available in packages like scikit-learn library [64], sktime [78] and GluonTS [79].

VII. RESULT AND DISCUSSION

In this section, we discuss the experimental results on real world financial time series. In the first part, we examine the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3068895, IEEE Access

TABLE 3. Performance metrics - Symmetric mean absolute percentage error (SMAPE), Root-mean-square error (RMSE) and Concordance index (CI) on equity indices and fixed income indices.

$N_{\text{train}} = 500$		$h = 50$			$h = 100$			$h = 200$		
$m = 100$		SMAPE	RMSE	CI	SMAPE	RMSE	CI	SMAPE	RMSE	CI
S&P 500	Theta Forecaster	0.02 (0.01)	55.86	0.55 (0.06)	0.03 (0.01)	81.21	0.51 (0.04)	0.04 (0.01)	110.72	0.51 (0.01)
	Linear Regression	0.02 (0.01)	53.01	0.64 (0.07)	0.03 (0.01)	84.53	0.60 (0.06)	0.05 (0.02)	129.65	0.61 (0.07)
	ARIMA	0.02 (0.01)	**51.12**	0.51 (0.02)	0.03 (0.01)	**73.58**	0.51 (0.01)	0.04 (0.02)	**107.55**	0.52 (0.03)
	DeepAR	0.03 (0.02)	58.10	0.52 (0.05)	0.06 (0.04)	86.55	0.52 (0.05)	0.06 (0.03)	143.26	0.53 (0.03)
	Random Forest	0.02 (0.01)	56.83	0.68 (0.14)	0.04 (0.01)	84.94	**0.72 (0.11)**	0.06 (0.02)	141.48	0.76 (0.14)
	LightGBM	-	-	0.62 (0.20)	-	-	0.72 (0.24)	-	-	0.77 (0.23)
	Logistic Regression	-	-	0.59 (0.17)	-	-	0.60 (0.18)	-	-	**0.80 (0.20)**
	Ordinal Regression	-	-	0.62 (0.20)	-	-	0.62 (0.20)	-	-	0.65 (0.22)
	Pairwise Ranking	-	-	0.63 (0.08)	-	-	0.61 (0.06)	-	-	0.62 (0.07)
	DynaPairwise Ranking	0.03 (0.02)	75.94	**0.73 (0.13)**	0.04 (0.01)	94.53	0.70 (0.15)	0.06 (0.02)	150.89	0.73 (0.14)
Russell 2000	Theta Forecaster	0.03 (0.01)	43.95	0.56 (0.06)	0.04 (0.02)	61.85	0.53 (0.05)	0.06 (0.02)	87.82	0.52 (0.04)
	Linear Regression	0.02 (0.01)	40.98	0.63 (0.09)	0.04 (0.02)	58.18	0.62 (0.05)	0.07 (0.03)	95.96	0.60 (0.11)
	ARIMA	0.03 (0.01)	**37.79**	0.53 (0.06)	0.04 (0.02)	**51.56**	0.54 (0.11)	0.06 (0.02)	**82.43**	0.55 (0.12)
	DeepAR	0.02 (0.02)	49.46	0.52 (0.06)	0.05 (0.03)	68.84	0.52 (0.04)	0.08 (0.4)	104.30	0.52 (0.01)
	Random Forest	0.03 (0.02)	46.26	0.66 (0.09)	0.05 (0.02)	62.80	0.68 (0.11)	0.07 (0.02)	92.78	0.67 (0.16)
	LightGBM	-	-	0.65 (0.21)	-	-	0.63 (0.20)	-	-	0.64 (0.22)
	Logistic Regression	-	-	0.61 (0.17)	-	-	0.65 (0.19)	-	-	**0.75 (0.22)**
	Ordinal Regression	-	-	0.59 (0.17)	-	-	0.63 (0.20)	-	-	0.70 (0.20)
	Pairwise Ranking	-	-	0.63 (0.08)	-	-	0.60 (0.08)	-	-	0.61 (0.10)
	DynaPairwise Ranking	0.04 (0.02)	57.99	**0.70 (0.11)**	0.05 (0.02)	73.30	**0.70 (0.11)**	0.09 (0.04)	124.69	0.65 (0.09)
DAX	Theta Forecaster	0.03 (0.02)	448.49	0.57 (0.06)	0.05 (0.02)	652.03	0.54 (0.04)	0.07 (0.03)	887.01	0.54 (0.02)
	Linear Regression	0.03 (0.01)	**410.94**	0.62 (0.08)	0.04 (0.02)	611.71	0.57 (0.05)	0.07 (0.03)	868.62	0.58 (0.07)
	ARIMA	0.03 (0.02)	419.74	0.54 (0.10)	0.05 (0.03)	**600.70**	0.53 (0.09)	0.07 (0.03)	**815.33**	0.57 (0.12)
	DeepAR	0.05 (0.02)	503.56	0.51 (0.06)	0.07 (0.04)	798.61	0.52 (0.04)	0.08 (0.04)	908.40	0.51 (0.03)
	Random Forest	0.04 (0.02)	427.67	0.68 (0.12)	0.06 (0.04)	650.77	0.68 (0.11)	0.08 (0.04)	882.51	0.71 (0.12)
	LightGBM	-	-	0.62 (0.19)	-	-	0.63 (0.21)	-	-	0.67 (0.19)
	Logistic Regression	-	-	0.60 (0.17)	-	-	0.63 (0.18)	-	-	**0.76 (0.17)**
	Ordinal Regression	-	-	0.54 (0.12)	-	-	0.57 (0.17)	-	-	0.57 (0.17)
	Pairwise Ranking	-	-	0.61 (0.07)	-	-	0.61 (0.06)	-	-	0.60 (0.08)
	DynaPairwise Ranking	0.05 (0.03)	721.36	**0.75 (0.10)**	0.06 (0.03)	830.78	**0.68 (0.14)**	0.09 (0.04)	1106.25	0.66 (0.16)
TOPIX	Theta Forecaster	0.04 (0.02)	71.39	0.60 (0.09)	0.06 (0.03)	111.43	0.58 (0.05)	0.12 (0.07)	192.67	0.58 (0.05)
	Linear Regression	0.03 (0.02)	67.03	0.64 (0.12)	0.06 (0.04)	108.99	0.62 (0.10)	0.12 (0.07)	189.64	0.63 (0.07)
	ARIMA	0.04 (0.03)	**63.98**	0.64 (0.17)	0.06 (0.03)	**102.09**	0.59 (0.13)	0.12 (0.06)	176.59	0.63 (0.14)
	DeepAR	0.05 (0.02)	83.52	0.53 (0.06)	0.08 (0.05)	133.58	0.51 (0.02)	0.09 (0.04)	**154.85**	0.51 (0.03)
	Random Forest	0.04 (0.02)	66.43	0.69 (0.11)	0.06 (0.04)	103.50	**0.74 (0.14)**	0.12 (0.06)	182.72	0.68 (0.17)
	LightGBM	-	-	0.66 (0.21)	-	-	0.61 (0.20)	-	-	**0.76 (0.23)**
	Logistic Regression	-	-	0.61 (0.17)	-	-	0.64 (0.21)	-	-	0.74 (0.17)
	Ordinal Regression	-	-	0.55 (0.15)	-	-	0.60 (0.18)	-	-	0.70 (0.23)
	Pairwise Ranking	-	-	0.66 (0.10)	-	-	0.63 (0.08)	-	-	0.64 (0.11)
	DynaPairwise Ranking	0.05 (0.03)	99.35	**0.70 (0.12)**	0.07 (0.03)	122.89	0.66 (0.10)	0.14 (0.05)	216.19	0.68 (0.11)
Bund 10Y	Theta Forecaster	0.01 (0.01)	2.96	0.56 (0.07)	0.01 (0.01)	4.19	0.53 (0.05)	0.02 (0.01)	**6.07**	0.51 (0.03)
	Linear Regression	0.01 (0.01)	2.97	0.64 (0.08)	0.02 (0.01)	4.48	0.62 (0.08)	0.02 (0.01)	6.55	0.61 (0.10)
	ARIMA	0.01 (0.01)	2.48	0.53 (0.06)	0.02 (0.01)	**3.73**	0.53 (0.08)	0.03 (0.02)	6.47	0.54 (0.08)
	DeepAR	0.01 (0.01)	**2.45**	0.51 (0.06)	0.02 (0.02)	4.35	0.51 (0.04)	0.04 (0.03)	9.24	0.51 (0.04)
	Random Forest	0.01 (0.01)	2.74	0.72 (0.13)	0.02 (0.01)	4.27	0.69 (0.15)	0.03 (0.02)	6.13	0.67 (0.18)
	LightGBM	-	-	0.61 (0.21)	-	-	0.61 (0.21)	-	-	0.57 (0.17)
	Logistic Regression	-	-	0.56 (0.15)	-	-	0.69 (0.21)	-	-	0.68 (0.21)
	Ordinal Regression	-	-	0.54 (0.12)	-	-	0.57 (0.16)	-	-	0.56 (0.15)
	Pairwise Ranking	-	-	0.64 (0.09)	-	-	0.56 (0.06)	-	-	0.56 (0.09)
	DynaPairwise Ranking	0.01 (0.01)	3.82	**0.72 (0.11)**	0.02 (0.01)	5.28	**0.69 (0.14)**	0.03 (0.01)	7.46	**0.70 (0.18)**
Gilt 10Y	Theta Forecaster	0.01 (0.01)	3.82	0.56 (0.07)	0.02 (0.01)	5.28	0.55 (0.08)	0.03 (0.02)	7.46	0.54 (0.09)
	Linear Regression	0.01 (0.01)	3.20	0.64 (0.05)	0.02 (0.01)	3.92	0.63 (0.07)	0.03 (0.02)	7.42	0.63 (0.11)
	ARIMA	0.01 (0.01)	3.10	0.55 (0.10)	0.02 (0.01)	**3.46**	0.52 (0.05)	0.02 (0.02)	**5.43**	0.52 (0.06)
	DeepAR	0.01 (0.01)	**2.97**	0.52 (0.05)	0.03 (0.02)	5.57	0.50 (0.04)	0.05 (0.03)	9.08	0.52 (0.02)
	Random Forest	0.02 (0.01)	3.30	**0.70 (0.11)**	0.02 (0.01)	4.26	0.67 (0.11)	0.03 (0.02)	6.19	0.65 (0.09)
	LightGBM	-	-	0.65 (0.22)	-	-	**0.69 (0.24)**	-	-	0.57 (0.17)
	Logistic Regression	-	-	0.58 (0.16)	-	-	0.64 (0.21)	-	-	**0.70 (0.23)**
	Ordinal Regression	-	-	0.60 (0.16)	-	-	0.60 (0.18)	-	-	0.63 (0.21)
	Pairwise Ranking	-	-	0.63 (0.08)	-	-	0.64 (0.11)	-	-	0.63 (0.13)
	DynaPairwise Ranking	0.02 (0.01)	4.45	0.68 (0.08)	0.02 (0.01)	5.49	0.66 (0.10)	0.04 (0.03)	9.26	0.60 (0.07)
Treasury 10Y	Theta Forecaster	0.01 (0.01)	2.38	0.61 (0.10)	0.01 (0.01)	3.22	0.60 (0.10)	0.02 (0.01)	4.37	0.58 (0.10)
	Linear Regression	0.01 (0.00)	**2.14**	0.65 (0.09)	0.01 (0.00)	**2.80**	0.63 (0.07)	0.01 (0.01)	3.79	0.60 (0.10)
	ARIMA	0.01 (0.00)	2.19	0.67 (0.16)	0.01 (0.01)	2.90	0.66 (0.18)	0.02 (0.01)	3.75	0.63 (0.17)
	DeepAR	0.01 (0.01)	2.22	0.52 (0.04)	0.02 (0.01)	4.54	0.52 (0.03)	0.04 (0.03)	10.02	0.51 (0.03)
	Random Forest	0.01 (0.01)	2.44	0.67 (0.11)	0.02 (0.01)	3.43	0.65 (0.09)	0.02 (0.01)	**3.70**	**0.70 (0.09)**
	LightGBM	-	-	0.64 (0.20)	-	-	0.64 (0.19)	-	-	0.50 (0.00)
	Logistic Regression	-	-	0.57 (0.15)	-	-	0.56 (0.11)	-	-	0.65 (0.21)
	Ordinal Regression	-	-	0.56 (0.14)	-	-	0.56 (0.15)	-	-	0.59 (0.16)
	Pairwise Ranking	-	-	0.65 (0.10)	-	-	0.63 (0.06)	-	-	0.58 (0.08)
	DynaPairwise Ranking	0.01 (0.01)	3.22	**0.70 (0.11)**	0.02 (0.01)	4.06	**0.66 (0.11)**	0.02 (0.01)	5.27	0.65 (0.11)

use of the proposed methods from an alternative perspective other than tracking error; specifically, we examine the impact of our proposed method from the lens of time to event (TTE), which is critical in certain financial applications [50], [51]. In the second part, we discuss the practical issues, such as the performance of the proposed methods on different types of time series and under different lengths of the forecast horizon. The details of the results are presented in Table 3 and Table 4.

A. IMPACTS OF CONCORDANCE INDEX

The CI is a well-established evaluation metric for survival analysis in medical statistics [80]. The metric addresses the differences along the timeline between a forecasted event and a ground truth event, which essentially indicates the time until the occurrence of an event of interest (EOI), e.g., death, onset of a disease, or failure of a machine, depending on the domain. The adoption of CI as the major evaluation metrics instead of the tracking errors, happens in a wide spectrum of applications from clinical research [70], epidemiology, and disease control [81] to predictive maintenance [2], reliability engineering [82], and insurance

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3068895, IEEE Access

TABLE 4. Performance metrics - Symmetric mean absolute percentage error (SMAPE), Root-mean-square error (RMSE) and Concordance index (CI) on currency time series.

$N_{\textbf{train}} = 500$		$h = 50$			$h = 100$			$h = 200$		
$m = 100$		**SMAPE**	**RMSE**	**CI**	**SMAPE**	**RMSE**	**CI**	**SMAPE**	**RMSE**	**CI**
EURUSD	Theta Forecaster	0.03 (0.02)	4.63	0.61 (0.08)	0.04 (0.03)	6.70	0.60 (0.10)	0.05 (0.02)	7.14	0.62 (0.10)
	Linear Regression	0.03 (0.02)	4.34	0.64 (0.08)	0.04 (0.03)	6.24	0.60 (0.06)	0.05 (0.02)	6.84	0.60 (0.06)
	ARIMA	0.03 (0.03)	3.99	0.56 (0.15)	0.05 (0.04)	6.13	0.57 (0.15)	0.07 (0.08)	9.10	0.62 (0.20)
	DeepAR	0.04 (0.03)	4.88	0.52 (0.05)	0.06 (0.04)	7.79	0.49 (0.04)	0.09 (0.08)	11.22	0.51 (0.03)
	Random Forest	0.03 (0.02)	**3.85**	0.66 (0.09)	0.04 (0.03)	**5.52**	0.64 (0.08)	0.04 (0.02)	**5.16**	0.64 (0.10)
	LightGBM	-	-	0.70 (0.22)	-	-	0.68 (0.23)	-	-	0.69 (0.22)
	Logistic Regression	-	-	0.59 (0.16)	-	-	0.69 (0.20)	-	-	0.75 (0.18)
	Ordinal Regression	-	-	0.62 (0.18)	-	-	0.67 (0.19)	-	-	0.53 (0.08)
	Pairwise Ranking	-	-	0.65 (0.08)	-	-	0.63 (0.09)	-	-	0.64 (0.07)
	DynaPairwise Ranking	0.03 (0.02)	4.86	**0.78 (0.14)**	0.04 (0.03)	6.66	**0.77 (0.18)**	0.05 (0.02)	7.35	**0.77 (0.12)**
GBPUSD	Theta Forecaster	0.03 (0.02)	4.21	0.65 (0.11)	0.04 (0.04)	6.78	0.62 (0.11)	0.04 (0.02)	7.10	0.63 (0.09)
	Linear Regression	0.02 (0.02)	4.02	0.67 (0.07)	0.04 (0.03)	6.25	0.63 (0.08)	0.04 (0.02)	6.65	0.60 (0.08)
	ARIMA	0.03 (0.02)	3.33	0.54 (0.10)	0.04 (0.04)	5.54	0.57 (0.13)	0.08 (0.10)	9.81	0.60 (0.16)
	DeepAR	0.03 (0.03)	3.57	0.51 (0.06)	0.05 (0.04)	7.00	0.52 (0.05)	0.10 (0.09)	11.68	0.51 (0.04)
	Random Forest	0.02 (0.02)	**3.13**	0.66 (0.09)	0.04 (0.03)	**5.29**	0.63 (0.10)	0.04 (0.02)	**5.33**	0.65 (0.10)
	LightGBM	-	-	0.71 (0.22)	-	-	0.73 (0.22)	-	-	0.83 (0.20)
	Logistic Regression	-	-	0.62 (0.19)	-	-	0.64 (0.20)	-	-	0.71 (0.20)
	Ordinal Regression	-	-	0.55 (0.13)	-	-	0.58 (0.17)	-	-	0.55 (0.14)
	Pairwise Ranking	-	-	0.65 (0.08)	-	-	0.64 (0.09)	-	-	0.63 (0.09)
	DynaPairwise Ranking	0.03 (0.02)	4.29	**0.76 (0.13)**	0.04 (0.03)	6.69	**0.79 (0.14)**	0.04 (0.02)	6.72	**0.86 (0.09)**
CHFUSD	Theta Forecaster	0.03 (0.02)	5.41	0.58 (0.07)	0.05 (0.04)	8.39	0.54 (0.38)	0.06 (0.03)	8.12	0.56 (0.07)
	Linear Regression	0.03 (0.03)	5.61	0.63 (0.09)	0.05 (0.04)	8.76	0.59 (0.09)	0.05 (0.03)	7.64	0.59 (0.07)
	ARIMA	0.03 (0.03)	4.33	0.56 (0.10)	0.06 (0.06)	8.02	0.53 (0.05)	0.06 (0.04)	7.86	0.54 (0.06)
	DeepAR	0.03 (0.03)	5.28	0.52 (0.07)	0.07 (0.04)	9.52	0.51 (0.04)	0.14 (0.12)	15.06	0.50 (0.02)
	Random Forest	0.03 (0.02)	**4.11**	0.69 (0.11)	0.05 (0.03)	**6.73**	0.65 (0.10)	0.06 (0.03)	**7.51**	0.65 (0.14)
	LightGBM	-	-	0.70 (0.21)	-	-	0.73 (0.20)	-	-	0.75 (0.20)
	Logistic Regression	-	-	0.62 (0.18)	-	-	0.67 (0.21)	-	-	0.73 (0.21)
	Ordinal Regression	-	-	0.66 (0.20)	-	-	0.68 (0.22)	-	-	0.75 (0.21)
	Pairwise Ranking	-	-	0.64 (0.07)	-	-	0.61 (0.08)	-	-	0.62 (0.09)
	DynaPairwise Ranking	0.03 (0.02)	5.63	**0.73 (0.13)**	0.05 (0.04)	8.36	**0.73 (0.18)**	0.05 (0.03)	8.13	**0.76 (0.19)**
JPYUSD	Theta Forecaster	0.02 (0.02)	2.77	0.63 (0.11)	0.03 (0.02)	4.06	0.59 (0.10)	0.05 (0.04)	6.73	0.57 (0.09)
	Linear Regression	0.02 (0.02)	2.76	0.62 (0.07)	0.03 (0.02)	4.03	0.57 (0.06)	0.05 (0.04)	6.78	0.56 (0.05)
	ARIMA	0.02 (0.02)	**2.45**	0.53 (0.07)	0.04 (0.02)	3.67	0.51 (0.02)	0.05 (0.04)	5.54	0.51 (0.03)
	DeepAR	0.03 (0.02)	3.12	0.54 (0.04)	0.04 (0.03)	3.90	0.49 (0.03)	0.14 (0.08)	13.75	0.51 (0.03)
	Random Forest	0.02 (0.02)	2.55	0.69 (0.11)	0.03 (0.02)	**3.59**	0.69 (0.13)	0.05 (0.04)	**5.27**	0.68 (0.13)
	LightGBM	-	-	0.67 (0.22)	-	-	0.65 (0.22)	-	-	0.68 (0.23)
	Logistic Regression	-	-	0.60 (0.18)	-	-	0.60 (0.17)	-	-	0.59 (0.14)
	Ordinal Regression	-	-	0.56 (0.13)	-	-	0.62 (0.19)	-	-	0.58 (0.16)
	Pairwise Ranking	-	-	0.62 (0.09)	-	-	0.60 (0.08)	-	-	0.60 (0.08)
	DynaPairwise Ranking	0.03 (0.02)	3.64	**0.73 (0.14)**	0.04 (0.02)	4.84	**0.83 (0.12)**	0.06 (0.04)	7.07	**0.83 (0.08)**
CADUSD	Theta Forecaster	0.02 (0.02)	5.13	0.63 (0.10)	0.03 (0.03)	6.73	0.60 (0.09)	0.03 (0.02)	7.38	0.57 (0.09)
	Linear Regression	0.02 (0.02)	4.92	0.63 (0.08)	0.03 (0.02)	6.19	0.61 (0.10)	0.03 (0.02)	6.44	0.57 (0.10)
	ARIMA	0.02 (0.02)	**3.67**	0.55 (0.11)	0.03 (0.03)	5.22	0.55 (0.13)	0.04 (0.02)	6.98	0.65 (0.21)
	DeepAR	0.02 (0.03)	4.04	0.52 (0.05)	0.04 (0.02)	6.60	0.51 (0.04)	0.09 (0.05)	16.16	0.50 (0.02)
	Random Forest	0.02 (0.02)	3.78	0.65 (0.09)	0.03 (0.02)	**5.22**	0.67 (0.12)	0.03 (0.02)	**6.38**	0.69 (0.11)
	LightGBM	-	-	0.65 (0.20)	-	-	0.62 (0.19)	-	-	0.68 (0.23)
	Logistic Regression	-	-	0.63 (0.19)	-	-	0.69 (0.21)	-	-	0.76 (0.20)
	Ordinal Regression	-	-	0.62 (0.18)	-	-	0.71 (0.22)	-	-	**0.87 (0.20)**
	Pairwise Ranking	-	-	0.64 (0.09)	-	-	0.61 (0.10)	-	-	0.57 (0.07)
	DynaPairwise Ranking	0.03 (0.02)	5.19	**0.78 (0.12)**	0.03 (0.02)	6.77	**0.78 (0.13)**	0.04 (0.05)	10.01	0.85 (0.11)
AUDUSD	Theta Forecaster	0.04 (0.04)	9.75	0.60 (0.07)	0.05 (0.05)	14.76	0.58 (0.06)	0.06 (0.04)	15.61	0.54 (0.05)
	Linear Regression	0.03 (0.03)	9.43	0.63 (0.10)	0.05 (0.05)	14.25	0.59 (0.08)	0.06 (0.04)	15.88	0.59 (0.08)
	ARIMA	0.04 (0.04)	**8.09**	0.60 (0.15)	0.07 (0.07)	13.40	0.59 (0.15)	0.11 (0.14)	21.88	0.56 (0.16)
	DeepAR	0.04 (0.04)	8.79	0.53 (0.06)	0.09 (0.06)	18.57	0.53 (0.04)	0.12 (0.05)	26.64	0.53 (0.02)
	Random Forest	0.04 (0.04)	8.17	0.70 (0.08)	0.06 (0.06)	**12.41**	0.68 (0.11)	0.07 (0.03)	**14.72**	0.70 (0.11)
	LightGBM	-	-	0.62 (0.19)	-	-	0.70 (0.21)	-	-	**0.77 (0.18)**
	Logistic Regression	-	-	0.57 (0.15)	-	-	0.55 (0.11)	-	-	0.55 (0.13)
	Ordinal Regression	-	-	0.58 (0.16)	-	-	0.63 (0.18)	-	-	0.69 (0.22)
	Pairwise Ranking	-	-	0.66 (0.11)	-	-	0.61 (0.10)	-	-	0.63 (0.09)
	DynaPairwise Ranking	0.04 (0.03)	11.42	**0.79 (0.13)**	0.06 (0.05)	15.72	**0.75 (0.14)**	0.07 (0.04)	17.97	0.71 (0.17)
NZDUSD	Theta Forecaster	0.04 (0.03)	9.32	0.58 (0.07)	0.05 (0.04)	14.17	0.55 (0.06)	0.07 (0.03)	17.03	0.53 (0.05)
	Linear Regression	0.03 (0.03)	8.44	0.64 (0.08)	0.05 (0.04)	13.16	0.59 (0.07)	0.07 (0.04)	16.80	0.62 (0.08)
	ARIMA	0.04 (0.03)	**8.16**	0.55 (0.11)	0.07 (0.05)	13.61	0.55 (0.12)	0.12 (0.13)	22.62	0.54 (0.11)
	DeepAR	0.04 (0.03)	9.10	0.52 (0.06)	0.06 (0.05)	13.50	0.51 (0.04)	0.20 (0.09)	39.06	0.50 (0.01)
	Random Forest	0.04 (0.04)	8.52	0.69 (0.12)	0.06 (0.05)	**12.49**	0.68 (0.14)	0.07 (0.02)	**15.64**	0.66 (0.09)
	LightGBM	-	-	0.59 (0.17)	-	-	0.68 (0.22)	-	-	0.76 (0.21)
	Logistic Regression	-	-	0.62 (0.19)	-	-	0.65 (0.19)	-	-	**0.77 (0.20)**
	Ordinal Regression	-	-	0.56 (0.14)	-	-	0.55 (0.13)	-	-	0.55 (0.14)
	Pairwise Ranking	-	-	0.66 (0.08)	-	-	0.63 (0.07)	-	-	0.60 (0.09)
	DynaPairwise Ranking	0.04 (0.03)	10.97	**0.75 (0.14)**	0.05 (0.04)	14.11	**0.78 (0.13)**	0.07 (0.03)	16.73	0.68 (0.17)
NOKUSD	Theta Forecaster	0.03 (0.03)	7.08	0.60 (0.09)	0.05 (0.04)	10.41	0.57 (0.08)	0.05 (0.02)	10.13	0.54 (0.05)
	Linear Regression	0.03 (0.03)	6.62	0.64 (0.09)	0.04 (0.03)	9.68	0.59 (0.07)	0.05 (0.02)	9.76	0.60 (0.08)
	ARIMA	0.03 (0.03)	5.83	0.60 (0.17)	0.05 (0.05)	9.12	0.62 (0.19)	0.08 (0.12)	12.63	0.74 (0.20)
	DeepAR	0.03 (0.03)	5.76	0.53 (0.06)	0.06 (0.04)	9.63	0.50 (0.03)	0.13 (0.11)	18.61	0.49 (0.02)
	Random Forest	0.03 (0.03)	**5.57**	0.66 (0.12)	0.05 (0.03)	**8.00**	0.64 (0.10)	0.05 (0.02)	**8.54**	0.63 (0.13)
	LightGBM	-	-	0.63 (0.20)	-	-	0.63 (0.19)	-	-	0.73 (0.23)
	Logistic Regression	-	-	0.65 (0.18)	-	-	0.74 (0.20)	-	-	0.77 (0.20)
	Ordinal Regression	-	-	0.62 (0.19)	-	-	0.61 (0.20)	-	-	0.59 (0.15)
	Pairwise Ranking	-	-	0.62 (0.09)	-	-	0.59 (0.07)	-	-	0.59 (0.06)
	DynaPairwise Ranking	0.04 (0.02)	7.53	**0.75 (0.13)**	0.06 (0.04)	11.93	**0.80 (0.14)**	0.07 (0.06)	15.24	**0.84 (0.10)**
SEKUSD	Theta Forecaster	0.04 (0.03)	5.56	0.59 (0.08)	0.05 (0.04)	7.95	0.57 (0.07)	0.06 (0.03)	8.82	0.56 (0.06)
	Linear Regression	0.03 (0.03)	5.15	0.62 (0.07)	0.04 (0.04)	7.37	0.58 (0.08)	0.05 (0.03)	8.11	0.58 (0.07)
	ARIMA	0.04 (0.03)	**4.71**	0.53 (0.10)	0.06 (0.05)	7.04	0.53 (0.09)	0.10 (0.13)	12.00	0.58 (0.15)
	DeepAR	0.04 (0.04)	5.41	0.51 (0.04)	0.08 (0.05)	10.25	0.51 (0.04)	0.09 (0.05)	11.50	0.48 (0.02)
	Random Forest	0.04 (0.03)	4.81	0.67 (0.10)	0.05 (0.04)	**6.22**	0.69 (0.11)	0.06 (0.04)	**7.51**	0.65 (0.12)
	LightGBM	-	-	0.64 (0.20)	-	-	0.67 (0.21)	-	-	0.70 (0.21)
	Logistic Regression	-	-	0.62 (0.19)	-	-	0.70 (0.20)	-	-	0.74 (0.22)
	Ordinal Regression	-	-	0.64 (0.20)	-	-	0.69 (0.21)	-	-	0.67 (0.20)
	Pairwise Ranking	-	-	0.63 (0.08)	-	-	0.59 (0.08)	-	-	0.59 (0.07)
	DynaPairwise Ranking	0.04 (0.03)	5.64	**0.75 (0.13)**	0.05 (0.04)	8.31	**0.80 (0.14)**	0.05 (0.03)	8.56	**0.82 (0.11)**

[83]. We hereby discuss its property, usage and potential impacts in financial applications.

Recalling the definition of the CI in Equation (19), TTE in Figure 4, and the gap between its ground truth and its forecast by the ranking model can be written as

$$\mathrm{TTE}(i) = \sum_{\substack{\forall r_j > r_i \\ \forall j > i}} \min(j-i) \tag{23}$$

$$\widehat{\mathrm{TTE}}(i) = \sum_{\substack{\forall \hat{r}_j > \hat{r}_i \\ \forall j > i}} \min(j-i) \tag{24}$$

$$\Delta\mathrm{TTE}(i) = \left|\mathrm{TTE}(i) - \widehat{\mathrm{TTE}}(i)\right| \tag{25}$$

where $\mathrm{TTE}(i)$ and $\widehat{\mathrm{TTE}}(i)$ refer to the length of the vertical black lines in Figure 4(a) and Figure 4(b) and a permutation of the neighboring items in the ground truth ranking list results in the same directional change of $\Delta\mathrm{TTE}(i)$ and CI. When the CI is maximized to 1, $\Delta\mathrm{TTE}(i)$ is minimized to 0. A higher CI score indicates not only the correctness of the relative ordering of the multi-step ahead forecast but also the similarity between the two TTE maps in Figure 4. In the financial domain where we conducted the experiment, the similarity of the ground truth TTE map and the forecasted TTE map is important. The existence of certain financial derivative instruments enable the monetization of the forecasted TTE map by our proposed models. Typical examples include future contracts, which enable the construction of a long-short investment portfolio [25], [84], [85]. With proper position sizing [86] and risk management, the signals or forecast derived from the model can be transformed into an investment strategy. Following the intuition that a future contract has a profit and loss profile that is dependent on the price difference at two different time epochs, we define the EOI as the successful observation of a higher value in the time series within a specified forward-looking horizon h from time i. With an accurate forecasting of the TTE map as depicted in Figure 4(b), investment decisions for hunting alternative returns from the financial asset class can be made easily. By using such forecasted TTE map, one can answer questions such as, should the current position be renewed at its termination, by how much and eventually convert them into systematic investment strategies.

The relationship between the CI and the TTE is intuitive. Figure 4 visualizes the TTE measurement of the ground truth and that of the forecasted relative ordering. The horizontal index indicates the starting time epoch of each event, and the vertical index indicates the time epoch when an EOI is first observed. The length of the black bars in the figure indicates the duration before an EOI occurs. The horizon of multi-step ahead forecasting is set to 50, and by forecasting 100 time epochs, we obtain a time-to-event estimate without censoring. In essence, the CI is a generalization of the gap between the ground truth TTE map in Figure 4(a) and the forecasted TTE map in Figure 4(b). Given a cross-temporal pair (i, j), TTE neglects the impact of the forecasted order during $t \in [i+1 : j-1]$, if they are correct with respect to $\hat{r}_t$, the forecasted order at time j.

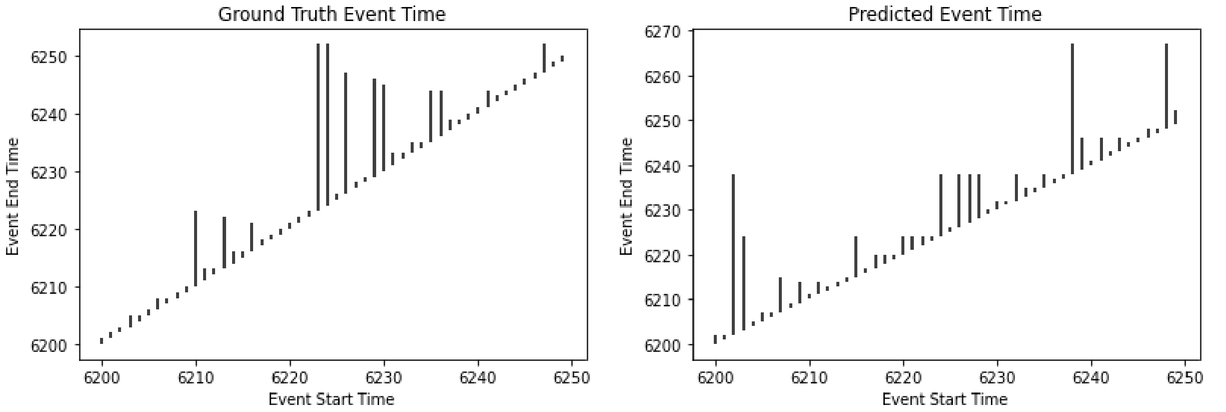
Overall, although it is difficult to directly optimize such a gap denoted by $\Delta\mathrm{TTE}(i)$, we argue that the proposed learning-to-rank approach which intends to optimize the concordance at a pairwise level is a promising solution for time-series forecasting, especially when relative ordering within a specified horizon matters to the domain experts.

B. GROWTH INDEX VS. MEAN-REVERTING INDEX

As shown in Table 3 and Table 4, the performance of the proposed method varies by the underlying asset class of the indices. For the growth indices in Table 3, we vary the forward-looking horizon of the competing models. In terms of our optimisation objective, i.e. CI, the proposed dynamic prediction scheme outperforms its competing methods in the short-term settings, where $h = 50$ and 100, and underperforms its competing methods in the long-term settings, where $h = 200$. Among the competing methods, the ARIMA model, the DeepAR model and the random forest model are the ones that deliver superior performance in terms of tracking error and deserve an attention. The DeepAR model outperformed consistently in fixed income indices, e.g. German 10Y Bund and UK 10Y Gilt. Given the low volatility of these fixed income indices recorded in Table 2, we think that this is because the DeepAR method captures the short-term trends well by leveraging the recurrent neural networks. However, the other side of this superior tracking error performance is its poor performance in terms of relative ordering or CI. Recalling that the growth indices share a common positive expected upward movement, we adopt Equation (22) to calibrate this expectation in reference to Table 2. However, such a calibration is trivial in short-term and does not show a significant impact in long-term forecasting with $h = 200$. We argue that this is partially due to the fact that the calibration is not integrated into the learning phase. When the length of the forward-looking horizon increases, the discriminative power of our model decreases dramatically; nevertheless, similar deterioration is also observed in the results by the competing methods.

In contrast to this significant deterioration, the performance deterioration with respect to the increase in the length of the forward-looking horizon is not significant for mean-reverting indices except for the ARIMA model and the DeepAR model, as observed in Table 4. We attribute this advantage of our proposed method to the increased volatility of the mean-reverting time series. The mean-reverting indices have on average higher standard deviations in terms of rolling return; therefore given a fixed length of horizon, both the features and the observations in the time series are more informative and discriminative. Since the momentum factor does not function well for mean-reverting time series, we also observed that among the competing methods, the random forest model performs consistently better than the ARIMA model and the DeepAR model. This is counterfactual to what we observed on the growth indices in Table 3 where the linear regression, the random forest model, the ARIMA model and the DeepAR model ties in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3068895, IEEE Access



(a) The ground truth time-to-event (TTE) map

(b) The forecasted time-to-event (TTE) map by the proposed method

FIGURE 4. The time-to-event (TTE) alignment between the ground truth and the forecast by the proposed method. The event of interest (EOI) is defined as the successful observation of a higher scalar. The horizontal index indicates the starting time epoch of each event, and the vertical index indicates the time epoch when an EOI is first observed. Therefore, the length of the black bars in the figure indicates the duration before an EOI occurs. The horizon of multi-step ahead forecasting is set to 50, and by forecasting 100 time epochs, we obtain a TTE estimate without censoring.

terms of tracking error metrics, e.g., SMAPE and RMSE. We argue that, for the ARIMA model, this is because the moving average pillar detracts the performance by explicitly incorporating the momentum or moving average for mean-reverting time series.

Overall, the proposed model outperforms its competing methods for all mean-reverting indices in almost all settings of the forward-looking horizon. This implies that the proposed dynamic prediction scheme is a better solution when the utility of the forecast is dependent on the relative ordering rather than on tracking error, and especially when the forecasting horizon is long and the time series exhibits high volatility or strong mean-reverting behavior.

Note that although the focus of this work is the relative ordering of multi-step ahead forecasting, armed with the LL algorithm described in Section IV, the proposed dynamic prediction scheme achieves comparable tracking error performance, e.g., SMAPE and RMSE, while significantly excelling its competing methods in terms of the CI. Interestingly, given a time series in Table 3 and Table 4, an improvement in the tracking error is not correlated with the improvement in the relative ordering metrics. We argue that, from this empirical observation and the discussion on TTE, the tracking error metrics and the relative ordering metrics that we advocate in our proposed method are orthogonal or even complementary. For example, the DeepAR model as one of the state-of-the-art method for time-series forecasting, reports a best RMSE score while records a worst CI score, like for German Bund 10Y and for UK Gilt 10Y. The two extreme cases of this kind are our proposed method with which the CI is optimised and the ARIMA method with which the tracking errors are optimised. It is important to note that the proposed method in this work is derived from a series of techniques that are not yet well explored in time-series forecasting problems. With the promising experimental results and the discovery as depicted above, we argue that the differences and connections between the competing methods and the proposed method worth a thorough investigation as our future works.

VIII. CONCLUSION

In this paper, we propose a learning-to-rank framework for multi-step ahead time-series forecasting which aims to optimise the relative ordering of the forecast. A local learning technique is incorporated in the framework to tackle the approximate inference of the time-series value given the forecast of its relative orderings at each time epoch along the timeline. A dynamic prediction scheme is proposed to integrate the proposed learning-to-rank model and the local learning in an iterative manner. Such a combination results in an improved performance in terms of the CI, which is the key evaluation metric for the relative ordering of the forecast. By comparing the proposed framework with a series of conventional methods on financial time series across different types of asset classes, we empirically verified that the proposed framework outperforms its competing methods from the lens of the CI. Finally, we conducted comprehensive examinations of the proposed method from multiple perspectives and elaborated the impact of the learning-to-rank model in time-series forecasting, the use of the proposed method, for different categories of time series and under different horizons of interest.

REFERENCES

[1] G. P. Zhang, "Time series forecasting using a hybrid arima and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[2] A. Kanawaday and A. Sane, "Machine learning for predictive maintenance of industrial machines using iot sensor data," in *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*. IEEE, 2017, pp. 87–90.

[3] K. Yamanishi and J.-i. Takeuchi, "A unifying framework for detecting outliers and change points from non-stationary time series data," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002, pp. 676–681.

[4] G. Koop and D. Korobilis, *Bayesian multivariate time series methods for empirical macroeconomics*. Now Publishers Inc, 2010.

[5] Y. Wolde-Rufael, "Electricity consumption and economic growth: a time series experience for 17 african countries," *Energy policy*, vol. 34, no. 10, pp. 1106–1114, 2006.

[6] R. S. Koijen and S. Van Nieuwerburgh, "Predictability of returns and cash flows," *Annu. Rev. Financ. Econ.*, vol. 3, no. 1, pp. 467–491, 2011.

[7] S. Gu, B. Kelly, and D. Xiu, "Empirical asset pricing via machine learning," *The Review of Financial Studies*, vol. 33, no. 5, pp. 2223–2273, 2020.

[8] J. D. Hamilton, "A new approach to the economic analysis of nonstationary time series and the business cycle," *Econometrica: Journal of the Econometric Society*, pp. 357–384, 1989.

[9] B. S. Bernanke and J. Boivin, "Monetary policy in a data-rich environment," *Journal of Monetary Economics*, vol. 50, no. 3, pp. 525–546, 2003.

[10] P. Krueger, Z. Sautner, and L. T. Starks, "The importance of climate risks for institutional investors," *The Review of Financial Studies*, vol. 33, no. 3, pp. 1067–1111, 2020.

[11] G. Bontempi, S. B. Taieb, and Y.-A. Le Borgne, "Machine learning strategies for time series forecasting," in *European business intelligence summer school*. Springer, 2012, pp. 62–77.

[12] X. Hong and S. Billings, "Time series multistep-ahead predictability estimation and ranking," *Journal of Forecasting*, vol. 18, no. 2, pp. 139–149, 1999.

[13] J. Contreras, R. Espinola, F. J. Nogales, and A. J. Conejo, "Arima models to predict next-day electricity prices," *IEEE transactions on power systems*, vol. 18, no. 3, pp. 1014–1020, 2003.

[14] G. E. Box and D. A. Pierce, "Distribution of residual autocorrelations in autoregressive-integrated moving average time series models," *Journal of the American statistical Association*, vol. 65, no. 332, pp. 1509–1526, 1970.

[15] H. Akaike, "Fitting autoregressive models for prediction," *Annals of the institute of Statistical Mathematics*, vol. 21, no. 1, pp. 243–247, 1969.

[16] R. L. Kashyap, "Optimal choice of ar and ma parts in autoregressive moving average models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 2, pp. 99–104, 1982.

[17] H. Tyralis and G. Papacharalampous, "Variable selection in time series forecasting using random forests," *Algorithms*, vol. 10, no. 4, p. 114, 2017.

[18] R. R. Yager, "Generalized dempster–shafer structures," *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 3, pp. 428–435, 2018.

[19] H.-K. Yu, "Weighted fuzzy time series models for taiex forecasting," *Physica A: Statistical Mechanics and its Applications*, vol. 349, no. 3-4, pp. 609–624, 2005.

[20] A. G. Parlos, O. T. Rais, and A. F. Atiya, "Multi-step-ahead prediction using dynamic recurrent neural networks," *Neural networks*, vol. 13, no. 7, pp. 765–786, 2000.

[21] Y. Bao, T. Xiong, and Z. Hu, "Multi-step-ahead time series prediction using multiple-output support vector regression," *Neurocomputing*, vol. 129, pp. 482–493, 2014.

[22] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, pp. 3146–3154, 2017.

[23] M. Cuturi and M. Blondel, "Soft-dtw: a differentiable loss function for time-series," in *International Conference on Machine Learning*, 2017, pp. 894–903.

[24] L. Vincent and N. Thome, "Shape and time distortion loss for training deep time series forecasting models," in *Advances in Neural Information Processing Systems*, 2019, pp. 4189–4201.

[25] B. Lim, S. Zohren, and S. Roberts, "Enhancing time-series momentum strategies using deep neural networks," *The Journal of Financial Data Science*, vol. 1, no. 4, pp. 19–38, 2019.

[26] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "Deepar: Probabilistic forecasting with autoregressive recurrent networks," *International Journal of Forecasting*, vol. 36, no. 3, pp. 1181–1191, 2020.

[27] R. J. Hyndman, R. A. Ahmed, G. Athanasopoulos, and H. L. Shang, "Optimal combination forecasts for hierarchical time series," *Computational statistics & data analysis*, vol. 55, no. 9, pp. 2579–2589, 2011.

[28] I. Moghram and S. Rahman, "Analysis and evaluation of five short-term load forecasting techniques," *IEEE Transactions on power systems*, vol. 4, no. 4, pp. 1484–1491, 1989.

[29] G. Liu and F. Xiao, "Time series data fusion based on evidence theory and owa operator," *Sensors*, vol. 19, no. 5, p. 1171, 2019.

[30] G. Liu, F. Xiao, C.-T. Lin, and Z. Cao, "A fuzzy interval time-series energy and financial forecasting model using network-based multiple time-frequency spaces and the induced-ordered weighted averaging aggregation operation," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 11, pp. 2677–2690, 2020.

[31] T. Xiong, Y. Bao, and Z. Hu, "Beyond one-step-ahead forecasting: evaluation of alternative multi-step-ahead forecasting models for crude oil prices," *Energy Economics*, vol. 40, pp. 405–415, 2013.

[32] V. Le Guen and N. Thome, "Probabilistic time series forecasting with shape and temporal diversity," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[33] M. Gönen and G. Heller, "Concordance probability and discriminatory power in proportional hazards regression," *Biometrika*, vol. 92, no. 4, pp. 965–970, 2005.

[34] S. Hayashi, A. Tanimoto, and H. Kashima, "Long-term prediction of small time-series data using generalized distillation," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.

[35] U. Herold and R. Maurer, "Structural positions and risk budgeting: Quantifying the impact of structural positions and deriving implications for active portfolio management," *Journal of Asset Management*, vol. 9, no. 2, pp. 149–157, 2008.

[36] S. B. Taieb, G. Bontempi, A. F. Atiya, and A. Sorjamaa, "A review and comparison of strategies for multi-step ahead time series forecasting based on the nn5 forecasting competition," *Expert systems with applications*, vol. 39, no. 8, pp. 7067–7083, 2012.

[37] S. B. Taieb and A. F. Atiya, "A bias and variance analysis for multistep-ahead time series forecasting," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 1, pp. 62–76, 2015.

[38] L. Lacasa, B. Luque, F. Ballesteros, J. Luque, and J. C. Nuno, "From time series to complex networks: The visibility graph," *Proceedings of the National Academy of Sciences*, vol. 105, no. 13, pp. 4972–4975, 2008.

[39] S. S. Rangapuram, M. W. Seeger, J. Gasthaus, L. Stella, Y. Wang, and T. Januschowski, "Deep state space models for time series forecasting," *Advances in neural information processing systems*, vol. 31, pp. 7785–7794, 2018.

[40] S. Mao and F. Xiao, "A novel method for forecasting construction cost index based on complex network," *Physica A: Statistical Mechanics and its Applications*, vol. 527, p. 121306, 2019.

[41] W. Bao, J. Yue, and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and long-short term memory," *PloS one*, vol. 12, no. 7, p. e0180944, 2017.

[42] X. Cheng, R. Zhang, J. Zhou, and W. Xu, "Deeptransport: Learning spatial-temporal dependency for traffic condition forecasting," in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–8.

[43] T. Chordia and L. Shivakumar, "Momentum, business cycle, and time-varying expected returns," *The Journal of Finance*, vol. 57, no. 2, pp. 985–1019, 2002.

[44] T. J. Moskowitz, Y. H. Ooi, and L. H. Pedersen, "Time series momentum," *Journal of financial economics*, vol. 104, no. 2, pp. 228–250, 2012.

[45] G. Koop, D. Korobilis, and D. Pettenuzzo, "Bayesian compressed vector autoregressions," *Journal of Econometrics*, vol. 210, no. 1, pp. 135–154, 2019.

[46] J.-K. Jung, M. Patnam, and A. Ter-Martirosyan, *An algorithmic crystal ball: forecasts-based on machine learning*. International Monetary Fund, 2018.

[47] K. Maehashi and M. Shintani, "Macroeconomic forecasting using factor models and machine learning: an application to japan," *Journal of the Japanese and International Economies*, vol. 58, p. 101104, 2020.

[48] N. Aunsri and P. Taveeapiradeecharoen, "A time-varying bayesian compressed vector autoregression for macroeconomic forecasting," *IEEE Access*, vol. 8, pp. 192 777–192 786, 2020.

[49] J. Che and J. Wang, "Short-term electricity prices forecasting based on support vector regression and auto-regressive integrated moving average modeling," *Energy Conversion and Management*, vol. 51, no. 10, pp. 1911–1917, 2010.

[50] A. S. Hall *et al.*, "Machine learning approaches to macroeconomic forecasting," *The Federal Reserve Bank of Kansas City Economic Review*, vol. 103, no. 63, p. 2, 2018.

[51] M. Nunez, R. Fidalgo-Merino, and R. Morales, "An event-based predictive modelling approach: An application in macroeconomics," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2018, pp. 1169–1174.

[52] P. G. Coulombe, M. Leroux, D. Stevanovic, and S. Surprenant, "How is machine learning useful for macroeconomic forecasting?" *arXiv preprint arXiv:2008.12477*, 2020.

[53] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip *et al.*, "Top 10 algorithms in data mining," *Knowledge and information systems*, vol. 14, no. 1, pp. 1–37, 2008.

[54] T. Minka, R. Cleven, and Y. Zaykov, "Trueskill 2: An improved bayesian skill rating system," *Tech. Rep.*, 2018.

[55] J. Duan, J. Li, Y. Baba, and H. Kashima, "A generalized model for multidimensional intransitivity," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2017, pp. 840–852.

[56] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li, "Learning to rank: from pairwise approach to listwise approach," in *Proceedings of the 24th international conference on Machine learning*, 2007, pp. 129–136.

[57] H.-T. Yu, A. Jatowt, H. Joho, J. M. Jose, X. Yang, and L. Chen, "Wassrank: Listwise document ranking using optimal transport theory," in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 2019, pp. 24–32.

[58] J. Baz, N. Granger, C. R. Harvey, N. Le Roux, and S. Rattray, "Dissecting investment strategies in the cross section and time series," *Available at SSRN 2695101*, 2015.

[59] N. Baltas and R. Kosowski, "Demystifying time-series momentum strategies: Volatility estimators, trading rules and pairwise correlations," *Market Momentum: Theory and Practice", Wiley*, 2020.

[60] T. Joachims, "A support vector method for multivariate performance measures," in *Proceedings of the 22nd international conference on Machine learning*, 2005, pp. 377–384.

[61] T. Joachims, "Training linear svms in linear time," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 217–226.

[62] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *the Journal of machine Learning research*, vol. 9, pp. 1871–1874, 2008.

[63] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*. Springer, 2010, pp. 177–186.

[64] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[65] L. Bottou, "Stochastic gradient descent tricks," in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 421–436.

[66] H. Zeng and Y.-m. Cheung, "Feature selection and kernel learning for local learning-based clustering," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 8, pp. 1532–1547, 2010.

[67] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," *Departmental Papers, University of Pennsylvania*, 2001.

[68] T. Lin, T. Guo, and K. Aberer, "Hybrid neural networks for learning the trend in time series," in *Proceedings of the twenty-sixth international joint conference on artificial intelligence*, 2017, pp. 2273–2279.

[69] C. Lemke and B. Gabrys, "Meta-learning for time series forecasting and forecast combination," *Neurocomputing*, vol. 73, no. 10-12, pp. 2006–2016, 2010.

[70] H. Steck, B. Krishnapuram, C. Dehing-Oberije, P. Lambin, and V. C. Raykar, "On ranking in survival analysis: Bounds on the concordance index," in *Advances in neural information processing systems*, 2008, pp. 1209–1216.

[71] P. Blanche, M. W. Kattan, and T. A. Gerds, "The c-index is not proper for the evaluation of-year predicted risks," *Biostatistics*, vol. 20, no. 2, pp. 347–357, 2019.

[72] V. Assimakopoulos and K. Nikolopoulos, "The theta model: a decomposition approach to forecasting," *International journal of forecasting*, vol. 16, no. 4, pp. 521–530, 2000.

[73] E. Spiliotis, V. Assimakopoulos, and S. Makridakis, "Generalizing the theta method for automatic forecasting," *European Journal of Operational Research*, vol. 284, no. 2, pp. 550–558, 2020.

[74] B. Kedem and K. Fokianos, *Regression models for time series analysis*. John Wiley & Sons, 2005, vol. 488.

[75] K.-Y. Liang and S. L. Zeger, "A class of logistic regression models for multivariate binary time series," *Journal of the American Statistical Association*, vol. 84, no. 406, pp. 447–451, 1989.

[76] H. Wu, A. Gattami, and M. Flierl, "Conditional mutual information-based contrastive loss for financial time series forecasting," *arXiv preprint arXiv:2002.07638*, 2020.

[77] M. Mohr, F. Wilhelm, M. Hartwig, R. Möller, and K. Keller, "New approaches in ordinal pattern representations for multivariate time series." in *FLAIRS Conference*, 2020, pp. 124–129.

[78] M. Löning, A. Bagnall, S. Ganesh, V. Kazakov, J. Lines, and F. J. Király, "sktime: A unified interface for machine learning with time series," *arXiv preprint arXiv:1909.07872*, 2019.

[79] A. Alexandrov, K. Benidis, M. Bohlke-Schneider, V. Flunkert, J. Gasthaus, T. Januschowski, D. C. Maddix, S. Rangapuram, D. Salinas, J. Schulz *et al.*, "Gluonts: Probabilistic and neural time series modeling in python," *Journal of Machine Learning Research*, vol. 21, no. 116, pp. 1–6, 2020.

[80] E. Longato, M. Vettoretti, and B. Di Camillo, "A practical perspective on the concordance index for the evaluation and selection of prognostic time-to-event models," *Journal of Biomedical Informatics*, p. 103496, 2020.

[81] A. Mayr and M. Schmid, "Boosting the concordance index for survival data–a unified framework to derive and evaluate biomarker combinations," *PloS one*, vol. 9, no. 1, p. e84483, 2014.

[82] S. Pölsterl, "scikit-survival: A library for time-to-event analysis built on top of scikit-learn," *Journal of Machine Learning Research*, vol. 21, no. 212, pp. 1–6, 2020.

[83] E. J. Sijbrands, E. Tornij, and S. J. Homsma, "Mortality risk prediction by an insurance company and long-term follow-up of 62,000 men," *PloS one*, vol. 4, no. 5, p. e5457, 2009.

[84] X. Hou, K. Wang, J. Zhang, and Z. Wei, "An enriched time-series forecasting framework for long-short portfolio strategy," *IEEE Access*, vol. 8, pp. 31 992–32 002, 2020.

[85] Z. Zhang, S. Zohren, and S. Roberts, "Deep reinforcement learning for trading," *The Journal of Financial Data Science*, vol. 2, no. 2, pp. 25–40, 2020.

[86] T. Spears, S. Zohren, and S. Roberts, "Investment sizing with deep learning prediction uncertainties for high-frequency eurodollar futures trading." *Available at SSRN 3664497*, 2020.

JIUDING DUAN received the B.S. degree from South China University of Technology, Guangzhou, China, in 2011, and the M.S. degree from Beihang University, Beijing, China, in 2014. He is a Ph.D. candidate in informatics at Kyoto University, Kyoto, Japan. Currently, he is associated with risklab at Allianz Global Investors GmbH after training assignments in Tokyo, London, Frankfurt, Munich and New York. Before joining Allianz Global Investors GmbH, he was a Research Assistant with the RIKEN Center for Advanced Intelligence Project from 2017 to 2018. His research interests include machine learning with structured data, Bayesian optimization, and their applications on ranking, time-series analysis and alternative risk premia strategy.

HISASHI KASHIMA is a professor at Department of Intelligence Science and Technology, Kyoto University. Before joining the faculty, he was a researcher at IBM Research - Tokyo from 1999 to 2009, and was an associate professor at Department of Mathematical Informatics, The University of Tokyo from 2009 to 2014. His research focuses on foundations of machine learning and data mining, and on their applications in various fields, including marketing, bio-/chemo-informatics, and industrial/business intelligence. He obtained his B.S. degree in applied mathematics and physics in 1997, and an M.S. degree in systems engineering in 1999, and a Ph.D. degree in informatics in 2007 from Kyoto University in Japan.