ML-assisted Optimization of Securities Lending

Abhinav Prasad
The Bank of New York Mellon
New York, NY, USA

Prakash Arunachalam
The Bank of New York Mellon
New York, NY, USA

Ali Motamedi
The Bank of New York Mellon
New York, NY, USA

Ranjeeta Bhattacharya
The Bank of New York Mellon
New York, NY, USA

Beibei Liu
The Bank of New York Mellon
New York, NY, USA

Hays Skip McCormick
The Bank of New York Mellon
New York, NY, USA

Shengzhe Xu
Computer Science
Virginia Tech
Arlington, VA, USA

Nikhil Muralidhar
Computer Science
Stevens Institute of Technology
Hoboken, NJ, USA

Naren Ramakrishnan
Computer Science
Virginia Tech
Arlington, VA, USA

ABSTRACT

This paper presents an integrated methodology to forecast the direction and magnitude of movements of lending rates in security markets. We develop a sequence-to-sequence (seq2seq) modeling framework that integrates feature engineering, motif mining, and temporal prediction in a unified manner to perform forecasting at scale in real-time or near real-time. We have deployed this approach in a large custodial setting demonstrating scalability to a large number of equities as well as newly introduced IPO-based securities in highly volatile environments.

CCS CONCEPTS

• **Computing methodologies** → **Neural networks**; • **Applied computing** → *Electronic commerce*; • **Information systems** → Data stream mining.

KEYWORDS

Securities Lending, Sequence-to-Sequence Modeling, Motif Mining, Deep Learning

ACM Reference Format:

Abhinav Prasad, Prakash Arunachalam, Ali Motamedi, Ranjeeta Bhattacharya, Beibei Liu, Hays Skip McCormick, Shengzhe Xu, Nikhil Muralidhar, and Naren Ramakrishnan. 2023. ML-assisted Optimization of Securities Lending. In *4th ACM International Conference on AI in Finance (ICAIF '23), November 27–29, 2023, Brooklyn, NY, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3604237.3626877

1 INTRODUCTION

Securities lending is an established practice in many custodial banks, including ours, wherein investment strategies backed by cash or non-cash collateral are enabled to support clients who wish to have additional returns on securities held. For example, the common method of shorting an equity or a fixed income security is to borrow the security and sell it. Later the short seller buys the security to return to the lender, profiting by any price decline net of lending fees. Those wishing to borrow or lend must search each other's offerings and negotiate a lending fee. Determining the optimal fee requires the lender to balance returns with utilization.

Furthermore, the process of determining the optimal fee for securities lending involves careful consideration of various factors. Lenders need to strike a balance between maximizing their returns and ensuring the efficient utilization of their securities. Setting the lending fee too high may discourage potential borrowers, leading to underutilization of the available securities. On the other hand, setting the fee too low might attract a high demand for borrowing, but the lender may not fully capitalize on the potential returns. It becomes crucial for lenders to assess market conditions, evaluate the demand for specific securities, and factor in the associated risks. By conducting thorough research and analysis, custodial banks can offer competitive lending fees that attract borrowers while optimizing the utilization of their securities. Additionally, establishing robust mechanisms for borrowers and lenders to connect and negotiate terms efficiently is paramount for the smooth functioning of securities lending operations. With effective fee determination strategies and streamlined processes, custodial banks can continue to provide valuable support to clients seeking enhanced returns on their securities holdings.

Why is this problem challenging? First, the complexity of over 100 demand and supply factors exacerbate the impact of even small adjustments to fees and rates. This complicates manual approaches to set rates for securities lending. Second, inconsistent patterns underlie the regimes under which securities lending can be profitable. When a security is in demand, lending fees tend to be very volatile. This presents a challenge to estimate how much the fee will rise or fall during a given period. Finally, the large number of securities with characteristic behaviors during different periods of time poses a scalability challenge. Overall, while digitization has supported operational efficiencies underlying securities lending, the above issues also have opened up the potential for errors which may cause cascading downstream effects. As a result, any approach to securities lending rate forecasting must contend with these uncertainties in order to deliver tangible results to operators.

ICAIF '23, November 27–29, 2023, Brooklyn, NY, USA

ACM ISBN 979-8-4007-0240-2/23/11.
https://doi.org/10.1145/3604237.3626877

1.1 Contributions

Our problem formulation here is to forecast both the direction and magnitude of lending fee movements. Our key contributions are three-fold.

• By forecasting magnitude along with the direction, our approach implicitly captures the nature of the security (e.g., general collateral, warm, special and hot). This suggests inputs to adjust the lending rate, especially when there is no market indication.

• We present a sequence-to-sequence (seq2seq) modeling framework that runs at scale in real-time to support securities lending in highly volatile environments.

• We outline applications and deployment in a large custodial setting demonstrating scalability to a large number of equities as well as newly introduced IPO-based securities.

1.2 Organization of this paper

Section 2 provides some background about securities lending. Section 3 surveys related research. Section 4 presents the overall methodology with example experimental results covered in Section 5. Finally, we present a discussion of the future implications of securities lending rate prediction in custodial markets.

2 BACKGROUND

As introduced earlier, the typical approach to short selling a stock or a fixed-income security involves borrowing the asset and subsequently selling it. Later, the short seller will repurchase the asset to return it to the lender, aiming to profit from any decrease in its price after deducting borrowing fees. Borrowers and lenders must find each other and agree on a fee through negotiation, which the lender sets to optimize utilization. When the collateral provided is non-cash, the borrower usually pays a fee, whereas for cash collateral, the borrower may receive a rebate, with the rebate rate determined by the lender (see Fig. 1).

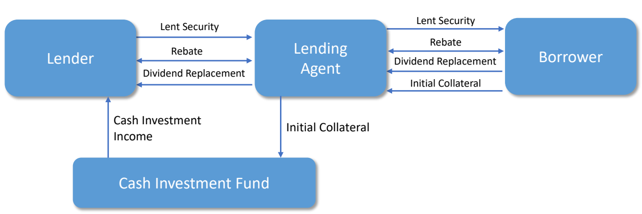


Figure 1: Overview of Securities-Lending Transaction Collateralized With Cash (adapted from Morningstar Analytics) [1]

Short selling has played a key role in recent market events and conditions. During the COVID-19 pandemic, the global financial markets experienced a severe downturn. As economic uncertainty grew, investors became increasingly concerned about the potential impact on various industries. In response, some investors engaged in short selling to profit from the anticipated decline in the stock prices of vulnerable companies. For instance, companies in the travel, hospitality, and oil sectors were heavily shorted as these industries faced significant challenges due to lockdowns and reduced consumer activity. Short sellers sought to capitalize on the price declines, which further added to the market volatility during that period.

As a second example, consider the GameStop Short Squeeze (2021). In early 2021, an extraordinary event unfolded in the stock market involving GameStop, a struggling video game retailer. A group of retail investors on a Reddit forum called *WallStreetBets* coordinated a massive buying campaign on GameStop's shares, causing its stock price to skyrocket. This sudden surge put significant pressure on hedge funds and institutional investors who had heavily shorted GameStop's stock, as they were facing massive losses. The short squeeze resulted in a rapid and unexpected upward movement in the stock's price, leading to considerable loss for short sellers and a notable shift in market dynamics.

To operationalize the securities lending situation here, we view the underlying business problem as one of increasing fee and rebate rate as close to the market rate, while keeping the utilization high and providing revenue uplift. The fee provided by the data is spread which contains volume weighted number for both cash collateral as well as non-cash collateral. Forecasts from our machine learning models are made available daily to the desk for repricing analysis. By identifying where the forecasted rate is in relation to the market mean and the securities relative volatility, combined with the directional and magnitude prediction, we obtain greater insights into potential rate movements.

3 RELATED RESEARCH

Forecasting models have long been employed for decision making in economics and financial sectors. Several modeling paradigms have been developed for modeling financial time-series. The paradigms can be broadly divided into model-based and data-driven approaches.

Model-based approaches. Such approaches for forecasting financial time-series are primarily developed by experts in quantitative econometric modeling and are based on established econometric theory. Such approaches employ significant domain knowledge and rely on domain experts to sift through and isolate the drivers of a process of interest. In such approaches, the process being modeled is usually decomposed using traditional time-series decomposition techniques [31] to isolate trends, seasonality and cycles following which auto-regressive approaches [12] (e.g., Vector-Autoregression, Autoregressive Integrated Moving Average) are employed to model the residual transition dynamics. Several approaches based on state-space models like Kalman Filters have also been employed for modeling commodity prices, process risk [28] and evolution of inflation [6]. However, many of these state-space modeling approaches require at least a partial knowledge of transition dynamics, are mostly linear (or locally non-linear e.g., Extended-Kalman Filter) and make strong assumptions about process distributions. Model-based approaches have been developed for forecasting lending, inflation and interest rates [5] and rely on statistical techniques to analyze historical lending rate data, identifying patterns and relationships with various market factors. For instance, researchers have explored the impact of supply and demand dynamics [2], market volatility [20], interest rates, and credit risk on lending rates. By incorporating these variables into econometric models, researchers have achieved varying degrees of success in forecasting lending rates accurately.

Data-Driven Approaches. With the recent trends in generation and availability of massive troves of process data and the development of sophisticated statistical and machine learning (ML), more recent econometric forecasting efforts have been geared towards leveraging the power of ML techniques. The advantage of these techniques is their ability to model more sophisticated (non-linear) functions with no domain-knowledge. These algorithms leverage advanced computational techniques to analyze large volumes of historical data and identify complex patterns that may be difficult to capture with traditional econometric models. Researchers have employed various Bayesian and frequentist methods from machine learning (ML) such as support vector machines [21], random forests to predict various processes like stock prices [25], security prices [38], and bank loan loss defaults [4]. Although several works [14, 18, 33, 36] employ machine learning solutions for stock price prediction, to the best of our knowledge, we are the first to address the problem of ML-assisted Optimization of Securities Lending.

Deep Learning for Time-Series Forecasting. Traditional data-driven techniques although flexible, require sophisticated feature-engineering and feature selection techniques to be successful. Recently, with the advent of affordable commodity GPUs, the paradigm of developing and training large deep neural network models has come to the fore. Deep learning (DL) pipelines have proved highly effective in various domains such as NLP [13], computer vision [8, 37], image generation [19], and time-series generation [9, 39], due to their ability to model complex functions and their ability to automatically learn representations through non-trivial compositions of raw data. DL models have also been widely adopted for various economic forecasting tasks [35]. Such approaches have shown effective performance in forecasting the evolution of financial processes [35] like prices of stocks, bonds, indices, commodities [41] and the price variations therein. Further, DL models have also been employed to forecast changes and evolution in interest rates [26]. In contrast to other modeling paradigms, DL pipelines in financial modeling contexts are able to learn from a wide range of input variables, including market indices, asset characteristics, and macroeconomic indicators, to improve the accuracy of their forecasts of a process of interest.

Many previous works have successfully employed motif mining to extract high level temporal behavior from data [29, 30]. Additionally, motif mining enables discretization of time series behavior, thereby alleviating any adverse effects of process noise [11]. In this work, we hence leverage time series discretization via motif mining and encode the temporal dynamics of the discretized time series using DL techniques (Long-short term memory networks [3, 7, 15, 17, 40]) for the task of forecasting lending rates.

4 METHODOLOGY

Security lending is a problem where we aim to optimize the *stock fee* to maximize our revenue by lending securities. Usually the evolution of the stock fee for a particular ticker is locally smooth from a temporal context. Specifically, if we consider a particular ticker τ, a stock fee at time t, y_t^τ is on average, strongly correlated with the stock fee values at time-steps $(t-1):(t-w)$ for some

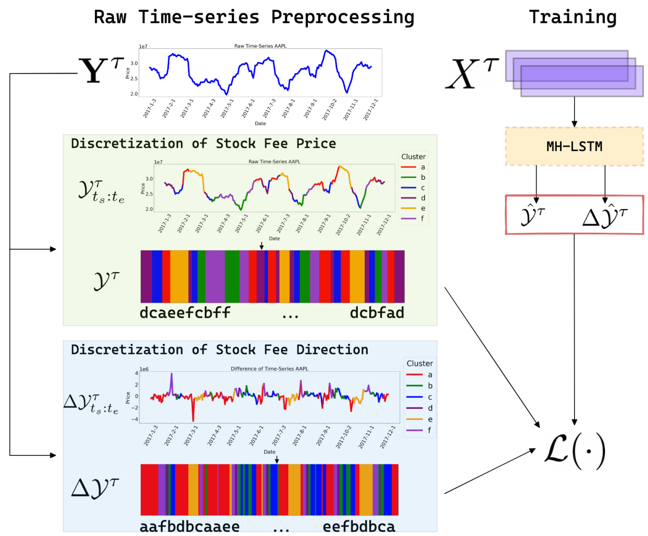


Figure 2: The architecture of our proposed multi-horizon LSTM model trained to estimate k-discretized stock fee price as well as stock fee direction of movement with categorical cross entropy and weak-Sobolev regularization.

local horizon 'w'. Taking advantage of this property of locally-deterministic temporal evolution of the *stock fee* process, we cast the problem of optimal stock fee estimation as a time-series forecasting problem. In general it is known that the evolution of the stock fee is also influenced by other factors like stock quantity and historical utilization.

Task Description. Let us denote the stock fee of a ticker τ over a particular horizon from start time t_s to end time t_e as $\mathbf{y}^\tau_{t_s:t_e} \in \mathbb{R}^{1\times h}$. Further, let us denote the corresponding values of exogenous variables (e.g., stock quantity and historical utilization) that are known to influence stock fee evolution as $\mathbf{e}^\tau_{t_s:t_e} \in \mathbb{R}^{q\times h}$. Here, h indicates the size of the time duration in days (i.e., $h = |t_e - t_s|$) and q is the number of exogenous factors considered to influence the stock fee. Our goal is to forecast the stock fee $\mathbf{y}^\tau_{t_s:t_e}$, from t_s to t_e for a ticker τ, conditioned upon historical information $X^\tau_{t_{s-w}:t_{s-1}} = [\mathbf{e}^\tau_{t_{(s-w)}:t_{(s-1)}}; \mathbf{y}^\tau_{t_{(s-w)}:t_{(s-1)}}]$. Here, $X^\tau_{t_{s-w}:t_{s-1}} \in \mathbb{R}^{(q+1)\times w}$ are the holistic set of input features for our proposed estimation task of stock fee for ticker τ in the horizon $t_s : t_e$. We employ a DL pipeline trained using empirical risk minimization (ERM) to model the evolution of our target task. Given, a training corpus of pairs of inputs, targets $\mathcal{X} = \{(X^\tau_{t_{s-w}:t_{s-1}}, \mathbf{y}^\tau_{t_s:t_e}) | w \le t_s \le M - h \wedge w + h \le t_e \le M\}$, Eq. 1 details the estimation objective.

$$\theta^\tau_{opt} = \underset{\theta\in\Theta}{\arg\min}\ \mathbb{E}_{(X^\tau,\mathbf{y}^\tau)\sim\mathcal{X}}[\mathcal{L}(\mathbf{y}^\tau, f_\theta(X^\tau))] \quad (1)$$

In Eq. 1, for each ticker τ, we learn a stock fee prediction function $f_\theta(\cdot)$ parameterized by learnable parameters θ and conditioned upon historical endogenous and exogenous inputs[1] X^τ. The ERM optimization objective in Eq. 1 yields, θ^τ_{opt}, which is the optimal

[1]we have dropped the temporal subscript notation for ease of understanding.

set of parameters that can be employed to yield estimates of stock fees in future time-steps.

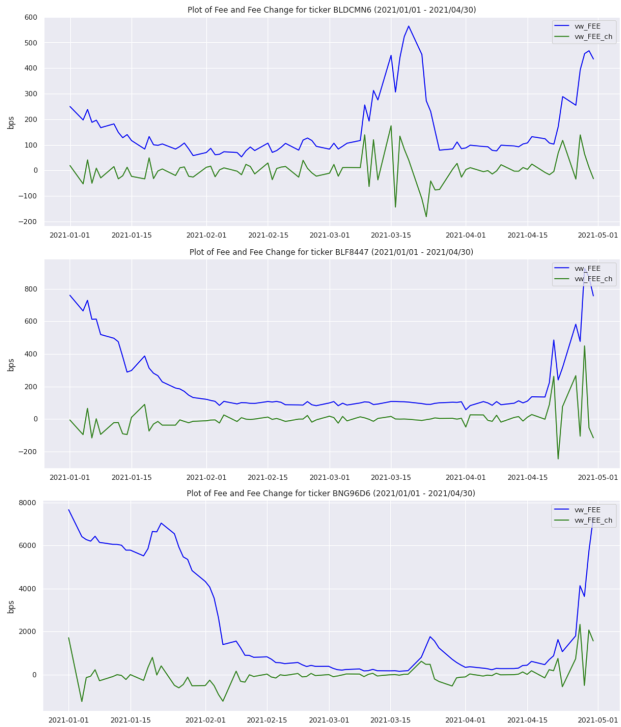
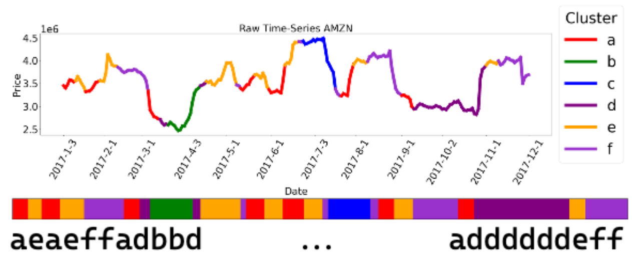


Figure 3: Showcase the stock fee (blue) and stock fee change (green) of various tickers. We notice that these processes exhibit complex temporal dynamics.

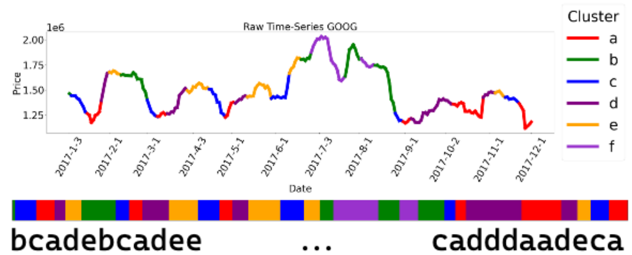
The stock fee of each ticker exhibits unique and complex evolutionary characteristics owing to the multivariate dependence on various exogenous factors. Fig. 3 demonstrate the uniqueness of evolutionary characteristics of the stock fee magnitude and the corresponding change of the stock fee price (i.e., stock fee direction) for different ticker symbols. Hence, attempting to model the stock fee with standard regression based DL models leads to sub-optimal predictions. To address this issue, we adopt a *discretized* modeling approach for the stock fee estimation task.

Discretization We employ a time-series discretization function $g : \mathbb{R}^{\mathcal{Y}} \rightarrow \mathbb{B}^{\mathcal{Y}}$ such that $\mathbb{R}^{\mathcal{Y}}$ is the continuous space comprising the raw stock fee data and $\mathbb{B}^{\mathcal{Y}}$ is the space of corresponding discretized representation.

For example (as Fig. 4), one instance of a simple discretization function could be one that discretizes a stock fee timeseries $\mathbf{Y}^{\tau}_{t_1:t_M} \in \mathbb{R}^{1\times M}$ for a particular stock ticker τ, based on *quartiles* (i.e., into a 4 symbol sequence) $\mathcal{Y}^{\tau}_{t_1:t_M} \in \mathbb{B}^{4\times M}$ where each time-step of the discretized sequence represents a 4-dimensional one-hot binary vector. In general we represent $\mathcal{Y}^{\tau}_{t_1:t_M} \in \mathbb{B}^{k\times M}$ as a *k-discretized* version of $\mathbf{y}^{\tau}_{t_1:t_M}$. Without loss of generality, we can consider the train-set $\mathcal{X}$ to comprise of input-target pairs $(X^{\tau}_{t_{s-w}:t_{s-1}}, \mathcal{Y}^{\tau}_{t_s:t_e})$ such that each target series $\mathcal{Y}^{\tau}_{t_s:t_e} \in \mathbb{B}^{k\times h}$ represents the *k-discrete* target series of the stock-fee $\mathbf{y}^{\tau}_{t_s:t_e} \in \mathbb{R}^{1\times h}$.



(a)



(b)

Figure 4: 4a - 4b showcase price time-series and their discretization $g(\cdot)$ in various stocks (AMZN, GOOG).

$$\theta^{\tau}_{opt} = \underset{\theta\in\Theta}{\arg\min}\ \mathbb{E}_{(X^{\tau},\mathcal{Y}^{\tau})\sim\mathcal{X}}[\mathcal{L}(\mathcal{Y}^{\tau}, f_{\theta}(X^{\tau}))] \tag{2}$$

Eq. 2 represents the variant of the ERM procedure outlined in Eq. 1 adapted for learning to estimate k-discretized stock fee time-series. The loss function $\mathcal{L}(\mathcal{Y}^{\tau}, f_{\theta}(X^{\tau}))$ in Eq. 2 plays a critical role in the estimation procedure and has been detailed in Eq. 3.

$$\mathcal{L}(\mathcal{Y}^{\tau}, f_{\theta}(X^{\tau})) = -\frac{1}{h}\sum_{i=1}^{h}\sum_{j=1}^{k}\mathcal{Y}^{\tau}_{ji}\log(\hat{\mathcal{Y}}^{\tau}_{ji}) \tag{3}$$

Eq. 3 demonstrates the categorical cross-entropy [16, 24] loss calculation over a target horizon h of a k-discretized target sequence $\mathcal{Y}^{\tau}$. Here, $\hat{\mathcal{Y}}\tau \in \mathbb{R}^{k\times h}$ represents the output of $f_{\theta}(X^{\tau})$. Essentially, each $\hat{\mathcal{Y}}^{\tau}_{ji} \in \mathcal{Y}^{\tau}$ represents the predicted probability (by $f_{\theta}(\cdot)$ conditioned on inputs X^{τ}) of occurrence of discrete symbol i at time sequence step j.

Weak Sobolev Regularization. Traditionally, DL pipelines make for an immensely powerful learning mechanism capable of modeling complicated functions. This capability can serve to be a doubled edged sword as it also brings with it the bane of *over-fitting* to the training data leading to dismal performance on unseen (test) data during inference, rendering the forecasting model useless. *Regularization* [34] has long been a popular technique in ML and DL employed to systematically curtail model complexity thereby reducing over-fitting. In addition to traditional (L1, L2) based regularization mechanisms, it has been recently shown that *Sobolev* losses [10] serve as effective regularizers in over-parameterized DL pipelines. We augment our training pipeline to incorporate a weak version of the Sobolev loss by training our DL pipeline to predict the k-discretized time-series of the gradient of the target time series i.e.,

$\Delta\mathcal{Y}^\tau = g(\Delta \mathrm{y}^\tau)$ where $g(\cdot)$ is a discretization function, $\Delta \mathrm{y}^\tau \in \mathbb{R}^{1\times h}$ is the first-difference (i.e. slope or *direction of motion*) at each point in the stock fee target series $\mathrm{y}^\tau \in \mathbb{R}^{1\times h}$ and $\Delta\mathcal{Y}^\tau \in \mathbb{B}^{k\times h}$ is the k-discretized version of the point-wise slope[2].

$$\theta^\tau_{opt} = \underset{\theta\in\Theta}{\arg\min}\ \mathbb{E}_{(X^\tau,\mathcal{Y}^\tau)\sim\mathcal{X}}[\mathcal{L}(\mathcal{Y}^\tau, f_\theta(X^\tau)) + \mathcal{L}(\Delta\mathcal{Y}^\tau, f_\theta(X^\tau))] \tag{4}$$

Our final training loss function with weak-Sobolev regularization in the discretized setting is detailed in Eq. 4.

4.1 Seq2seq Model Architecture

In an effort to encode the temporal nature of the target process of interest, we shall adopt recurrent deep learning architectures and a sequence-to-sequence forecasting design [23]. Our choice of modeling pipeline is an LSTM (long-short-term memory) network, which is adept at processing sequential data.

$$\begin{aligned} i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \\ f_t &= \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \\ c_t &= f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \\ o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \\ h_t &= o_t \tanh(c_t) \end{aligned} \tag{5}$$

Eq. 5[3] details the multiple gating mechanisms existing within an LSTM network. Each W matrix represents a learnable set of projections of the input sequence $x \in \mathbb{R}^{(q+1)\times w}$, hidden representations h_{t-i} or cell-states c_{t-i}. By maintaining separate states (i.e., the hidden states and cell states) as well as mechanisms (termed *gates*) to include and forget (f_t) information, LSTMs have been successful in overcoming the vanishing gradient [27] problems inherent to training vanilla Recurrent-neural networks (RNN).

The structure of our proposed LSTM pipeline is visually depicted in Fig. 5. Our proposed architecture comprises of two separate LSTM *trunk* models, each with the same structure of blocks. Each trunk net is designed to accept a different historical sequence length (one trunk accepts a 5 sequence input while the other accepts a 12 sequence input), thereby serving to incorporate *multi-horizon* effects into our modeling pipeline.

The architecture for both the LSTM models is identical – One input layer, One GroupNormalization layer, two LSTM layers (stacked LSTM) where each layer is followed by dropout layer and one Dense Layer on which ELU activation function is applied. The output from the models are combined using Average Layer where features generated during training from both the models are averaged and fed to another dense layer in order to generate the final prediction. Further, our model yields predictions to simultaneously forecast the k-discretized magnitude ($\mathcal{Y}^\tau$) and direction ($\Delta\mathcal{Y}^\tau$) of the stock fee for ticker τ. The activation function applied on the final dense layer is a softmax function. The loss function selected for optimization is

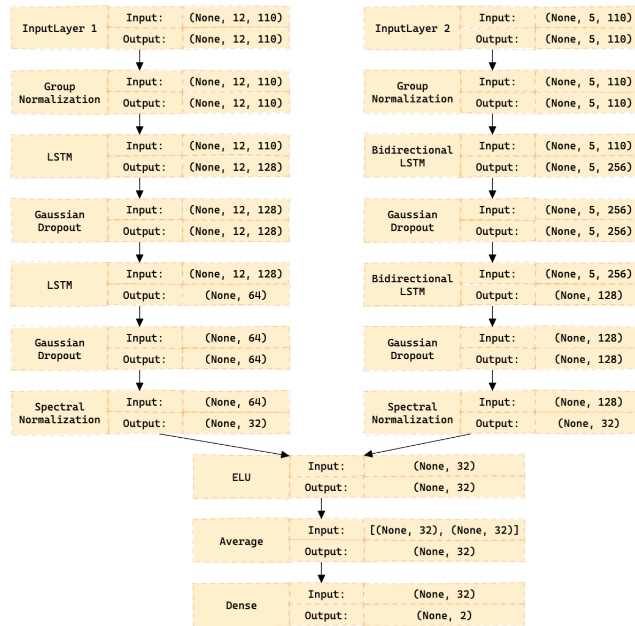


Figure 5: Proposed MH-LSTM stock fee forecasting model architecture with weak Sobolev regularization. The *None* in each layer structure indicates a flexible input-dimension of the input tensor to the layer that is set during execution e.g., batch size.

categorical cross-entropy (see Eq. 3) and the optimizer selected is Adam.

Finally, we detail a feature normalization technique incorporated to ensure increased stability and robustness during training of our DL pipeline. It is well-known that initial values of the weights of a neural network can have a significant impact on the training process. Weights should be chosen randomly but in a way to ensure appropriate activation in the linear region of activation functions like the sigmoid to avoid *saturation* [32] and consequent sub-optimal training. Works like [22] have investigated the effect on vanishing gradient on neural network training and proposed techniques to alleviate the ill-effects of vanishing gradients gradients during training by intelligent weight initialization and weight scaling.

Owing to the critical nature of this problem, we employ a technique based on Gaussian-Mixture-Models (GMM) to fit a feature distribution for each feature and employ the means and standard deviations of the learned GMM distribution to normalize and scale features in the neural network pipeline. Hyperparameters like the number of Gaussians in the GMM are obtained based on an elbow-curve method using a Bayesian-Information Criterion.

We report the effect of the GMM based feature-scaling procedure on the prediction performance of MH-LSTM in Table 1 (without GMM based feature scaling) and Table 2 (with GMM based feature scaling). We notice that incorporating feature scaling leads to increase in overall performance accuracy.

[2] For simplicity we consider both $\Delta\mathcal{Y}^\tau$ and $\mathcal{Y}^\tau$ to be k-discretized, in reality they don't have to be discretized into the same number of 'k' symbols.

[3] Note, here we have used overloaded notation to contextualize the LSTM mechanism for correspondence with standard literature. This must not be confused with terminology in Section 4.

Table 1: Forecasting results for our proposed modeling pipeline before GMMN based normalization.

	Precision	Recall	F1-Score
0	0.70	0.89	0.78
1	0.77	0.50	0.61
Accuracy			0.72
Macro Avg	0.73	0.69	0.69
Weighted Avg	0.73	0.72	0.71

Table 2: Forecasting results employing our proposed modeling pipeline, after GMMN based normalization. We notice an increase in the accuracy, macro and weighted average metrics, relative to the performance of the variant without GMMN (see Table. 1).

	Precision	Recall	F1-Score
0	0.76	0.74	0.75
1	0.70	0.71	0.70
Accuracy			0.73
Macro Avg	0.73	0.73	0.73
Weighted Avg	0.73	0.73	0.73

5 EXPERIMENTAL RESULTS

We perform a holistic performance evaluation by investigating the performance of MH-LSTM on the direction prediction and the magnitude prediction for the target task of stock fee estimation on a dataset of US-based securities. Further, we investigate the model performance on an out-of-sample time-period in April 2022.

5.1 Direction Prediction Task

In order to demonstrate the performance of our proposed model on the stock fee direction prediction task, we first investigate the training loss curve (see Fig. 6) of our proposed multi-horizon LSTM pipeline. We notice that the convergence as depicted, increases and becomes more reliable as the number of training epochs increases. Further, Table 3 details the task performance of the model on the out-of-sample data. We see from the table that our trained model yields an average accuracy of 68% on the task of direction prediction, across the out-of-sample period, on the investigated dataset of US-based securities. Further, we discovered that calibrating the classification threshold to 60% (from its original balanced threshold value of 50%) yields a significant rise in classification accuracy with the new accuracy reaching 75% on the binary classification task of daily stock fee direction prediction. Overall, the table indicates consistent prediction performance of our proposed multi-horizon LSTM pipeline, across all the classes (0, 1). From the table, we glean that the overall accuracy over out-of-sample period is 68%. We further investigate the distribution of the accuracy across the evaluation dataset (containing data from 6500 tickers) and report the histogram in Fig. 7. We notice that a majority of the tickers (70% or more) are above 60% in accuracy mainly in 80% to 100% accuracy range. Although there exist a few tickers with individual accuracy less than 60%, as shown by the distribution plot, the predictions with accuracy higher can be used for decision making.

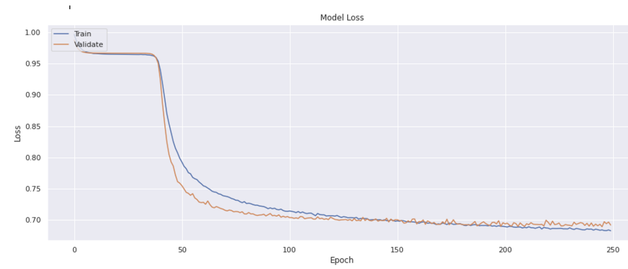


Figure 6: Loss Curve of direction prediction task in proposed multi-horizon LSTM model

Table 3: Performance of multi-horizon LSTM model on the direction prediction task for US-based securities on the out-of-sample test set (April 15^{th} - 22^{nd} 2022). We obtain an average classification accuracy of 68% with balanced classification threshold (0.5).

	Precision	Recall	F1-Score	Support
0	0.68	0.71	0.70	16917
1	0.68	0.65	0.67	16123
Accuracy			0.68	33040
Macro Avg	0.68	0.68	0.68	33040
Weighted Avg	0.68	0.68	0.68	33040

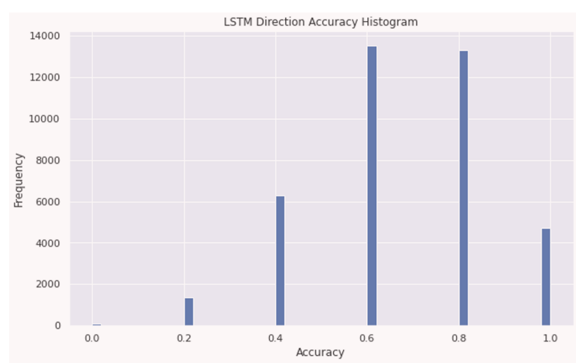


Figure 7: MH-LSTM stock fee direction prediction accuracy histogram on out-of-sample period on US-securities dataset. We notice that a majority of the predictions for the direction prediction by MH-LSTM have high-accuracy i.e., our model demonstrates greater than 60% accuracy on a majority of tickers for the stock fee direction prediction task.

Fig. 8 and Fig. 9 show that MH-LSTM yields highly accurate results for a majority of its decisions on the stock fee direction prediction task. We observe that although the model can be less than 60% accurate during a few time periods (and for a few tickers),

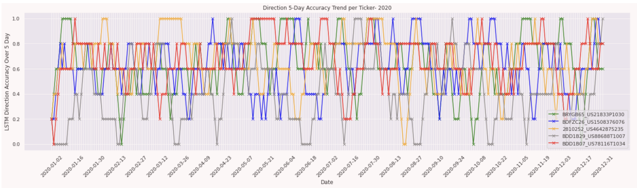


Figure 8: Long-term evolution across the entire year 2020 of 5-day-ahead stock fee direction prediction accuracy. The plot showcases the change in accuracy for the 5-day-ahead stock direction prediction task by MH-LSTM for 5 sample tickers. We notice that MH-LSTM yields prediction accuracies in excess of 60% for a majority of the direction prediction decisions.

it demonstrates decision accuracy in excess of 60% a vast majority of the time. This consistency and stability in prediction performance can be leveraged to consider predictions.

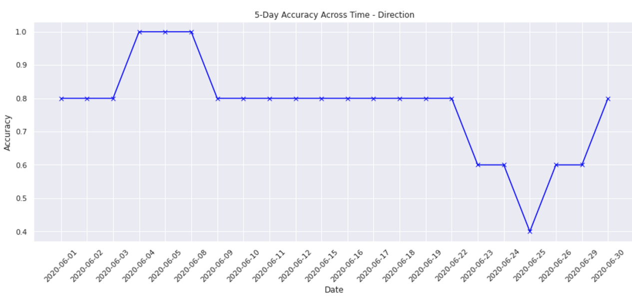


Figure 9: We isolate a sample ticker from Fig. 8 and depict (for a single month) a more granular evolution of 5-day prediction accuracy for the task of stock direction prediction by MH-LSTM . The plot depicts that MH-LSTM yields consistent and high accuracy (greater than or equal to 80%) for a majority of the decisions in the direction prediction task.

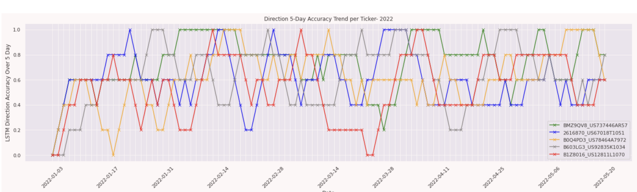


Figure 10: 5-day accuracy calculated everyday over 2022 prediction by MH-LSTM for the task of direction prediction on 5000 tickers.

Although a few regions in Fig. 10 showcase that MH-LSTM can yield prediction performance accuracy lower than 60%, we see that for a majority of the decisions, it is highly accurate and yields accuracy greater than 60%. Finally, we show the performance of tickers that are above a fixed threshold of 60%, in Fig. 11.

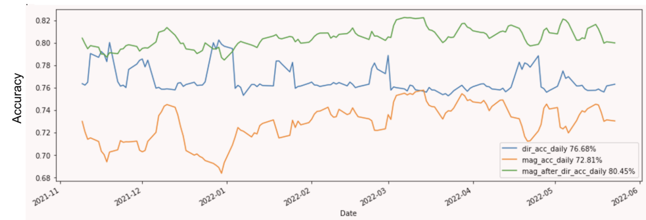


Figure 11: The performance of tickers that are above a fixed threshold of 60%.

5.2 Magnitude Prediction Task

In this section we further investigate the performance of MH-LSTM on the out-of-sample set from April 2022.

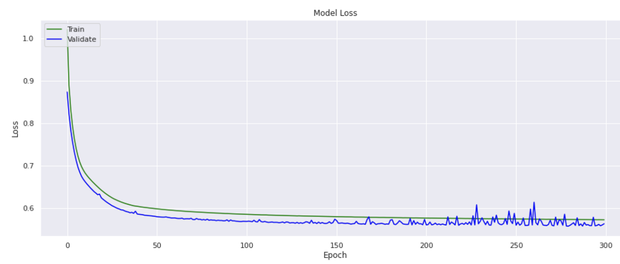


Figure 12: Learning Curve for magnitude prediction task of MH-LSTM .

We once again showcase the training and validation losses of MH-LSTM on the magnitude prediction task (Fig. 12) and observe a simultaneous decreasing trend in training and validation losses indicating a stable training of MH-LSTM for the task.

Further, we once again calculate the precision/recall/F1 score of MH-LSTM on the 5-day stock fee magnitude prediction task and find that the overall accuracy of MH-LSTM on the stock fee magnitude prediction task is 73% for 6500 US tickers. The classification results, as shown in Table 4, is also based on out-of-sample test set of April 2022.

Table 4: Performance of MH-LSTM model on the magnitude prediction task for US-based securities on the out-of-sample test set (April 15th - 22nd 2022). We obtain an average classification accuracy of 73% with balanced classification threshold.

	Precision	Recall	F1-Score	Support
0	0.76	0.74	0.75	17410
1	0.70	0.61	0.70	14535
Accuracy			0.73	31945
Macro Avg	0.73	0.73	0.73	31945
Weighted Avg	0.73	0.73	0.73	31945

Fig. 13 is the accuracy histogram of all tickers and it is also based on out of sample data on same period (2021/04/23 – 2021/04/30). Fig. 14 shows that majority of the tickers are in 60% to 100% accuracy range in terms of individual accuracy.There will be tickers with individual accuracy less than 60% as shown by the distribution plot but the predictions with accuracy higher than 60% may be selected to make better decisions. Fig. 15 shows 5-day accuracy calculated for magnitude model: 500 tickers (everyday over one year (2020)).

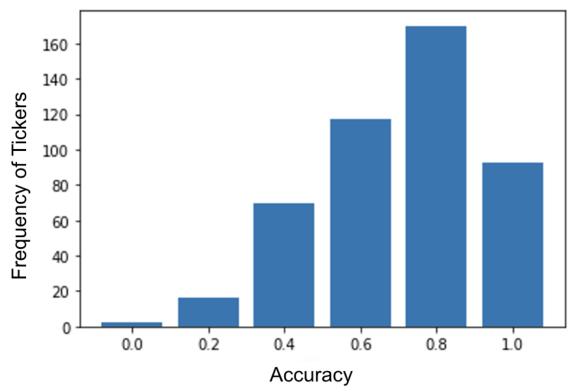


Figure 13: Accuracy histogram of all tickers. Y-axis is the frequency of tickers, where x-axis is the accuracy.

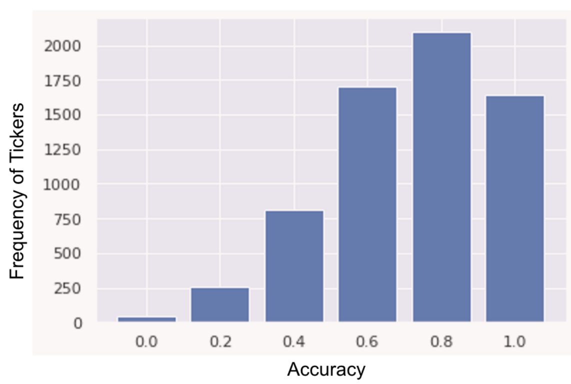


Figure 14: Accuracy histogram of all tickers. Y-axis is the frequency of tickers, where x-axis is the accuracy.

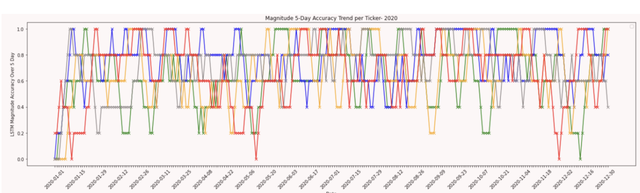
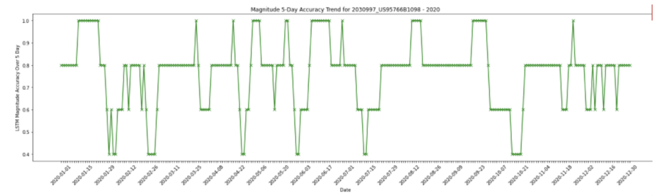
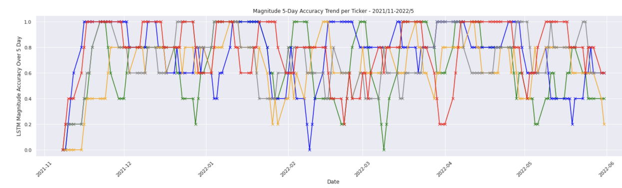


Figure 15: Accuracy evolution of MH-LSTM predictions for stock fee magnitude prediction task.

The 5-day accuracy plot for individual ticker is also stable and can be visualized using Fig. 16a, Fig. 16b.
Overall, our results demonstrate that MH-LSTM is able to yield stable and accurate predictions across diverse time ranges and tickers for the stock fee magnitude and direction prediction tasks.



(a) Magnitude 5 - Day Accuracy Trend for 2030997_US59576681098 in 2020



(b) Magnitude 5 - Day Accuracy Trend per Ticker from 11/2021 to 05/2022

Figure 16: Figures (a), (b) depict the MH-LSTM 5-day prediction accuracy for the stock fee magnitude prediction task for sample tickers.

6 CONCLUSION & FUTURE WORK

In this work, we have developed a scalable and integrated methodology based on a novel deep-learning pipeline for forecasting lending rates in security markets. Our flexible framework MH-LSTM , is able to model both the price and the direction of evolution of the lending rates making for a more holistic forecast. We have demonstrated the accurate and high-quality forecasts through rigorous qualitative and quantitative analyses employing 6500 tickers from the US equity market thereby demonstrating that our model is able to capture a large number of versatile market trends. Further, by developing our MH-LSTM model, scalability of the model to a large number of time series is an important problem that we have addressed. The benefit is double sided where we not only have a model that benefits from large dataset but also simplifies training, inference and process monitoring. Through our experiments we have learnt (and communicated) detailed and valuable lessons on the effectiveness of specialized normalization techniques for LSTM based neural networks.

In the future, we plan to augment our flexible DL pipeline to incorporate exogenous information based on market sentiment. Further, we shall also augment our DL based pipeline to incorporate the importance of market connectivity and provided insights into the dynamics of lending rate formation. Specifically, we shall incorporate the relationships and inter-connectedness between borrowers, lenders, and other market participants in the securities lending market. By analyzing the effects of the network structure on forecasting prowess, key influencers and nodes that play a crucial role in determining lending rates can be identified.

ACKNOWLEDGMENTS

We are grateful for comments and suggestions from Richard Marquis, William Kelly, Simon Tomlinson, Dave DiNardo, Christina Mackrell, and the Markets team at BNY Mellon. We are grateful to

Nikunj Bhalla for the engineering work and integration of model. We are also grateful to Eoin Lane for all the support and engagements and the Advanced solutions team. Finally, we are grateful for the critical leadership and support from Michael Demissie through the entire project timeline.

REFERENCES

[1] 2023. https://www.morningstar.com/etfs/understanding-securities-lending-etfs

[2] Carlo Altavilla, Miguel Boucinha, and Paul Bouscasse. 2022. Supply or demand: What drives fluctuations in the bank loan market? (2022).

[3] Wei Bao, Jun Yue, and Yulei Rao. 2017. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PloS one* 12, 7 (2017), e0180944.

[4] Joao A Bastos. 2010. Forecasting bank loans loss-given-default. *Journal of Banking & Finance* 34, 10 (2010), 2510–2517.

[5] SM Husnain BOKHARI and Mete Feridun. 2006. Forecasting inflation through econometric models: An empirical study on Pakistani data. *Doğuş Üniversitesi Dergisi* 7, 1 (2006), 39–47.

[6] Edwin Burmeister, Kent D Wall, and James D Hamilton. 1986. Estimation of unobserved expected monthly inflation using Kalman filtering. *Journal of Business & Economic Statistics* 4, 2 (1986), 147–160.

[7] Bowen Cai. 2021. Deep learning-based economic forecasting for the new energy vehicle industry. *Journal of Mathematics* 2021 (2021), 1–10.

[8] Junyi Chai, Hao Zeng, Anming Li, and Eric WT Ngai. 2021. Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Machine Learning with Applications* 6 (2021), 100134.

[9] Hongjie Chen, Ryan A Rossi, Kanak Mahadik, Sungchul Kim, and Hoda Eldardiry. 2023. Graph Deep Factors for Probabilistic Time-series Forecasting. *ACM Transactions on Knowledge Discovery from Data* 17, 2 (2023), 1–30.

[10] Jorio Cocola and Paul Hand. 2020. Global Convergence of Sobolev Training for Overparameterized Neural Networks. In *Machine Learning, Optimization, and Data Science: 6th International Conference, LOD 2020, Siena, Italy, July 19–23, 2020, Revised Selected Papers, Part I 6*. Springer, 574–586.

[11] A Davidović, EH Huntington, and MR Frater. 2009. Discretization in time gives rise to noise-induced improvement of the signal-to-noise ratio in static nonlinearities. *Physical Review E* 80, 1 (2009), 011119.

[12] Jan G De Gooijer and Rob J Hyndman. 2006. 25 years of time series forecasting. *International journal of forecasting* 22, 3 (2006), 443–473.

[13] Li Deng and Yang Liu. 2018. *Deep learning in natural language processing*. Springer.

[14] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. 2015. Deep learning for event-driven stock prediction. In *Twenty-fourth international joint conference on artificial intelligence*.

[15] Thomas Fischer and Christopher Krauss. 2018. Deep learning with long short-term memory networks for financial market predictions. *European journal of operational research* 270, 2 (2018), 654–669.

[16] Elliott Gordon-Rodriguez, Gabriel Loaiza-Ganem, Geoff Pleiss, and John Patrick Cunningham. 2020. Uses and abuses of the cross-entropy loss: Case studies in modern deep learning. (2020).

[17] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[18] Zexin Hu, Yiqi Zhao, and Matloob Khushi. 2021. A survey of forex and stock price prediction using deep learning. *Applied System Innovation* 4, 1 (2021), 9.

[19] Licheng Jiao and Jin Zhao. 2019. A survey on the new generation of deep learning in image processing. *Ieee Access* 7 (2019), 172231–172263.

[20] Jungsuk Kim, Abhishek Kumar, Sushanta Mallick, and Donghyun Park. 2021. Financial uncertainty and interest rate movements: is Asian bond market volatility different? *Annals of Operations Research* (2021), 1–29.

[21] Akshit Kurani, Pavan Doshi, Aarya Vakharia, and Manan Shah. 2023. A comprehensive comparative study of artificial neural network (ANN) and support vector machines (SVM) on stock forecasting. *Annals of Data Science* 10, 1 (2023), 183–208.

[22] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[23] Nikhil Muralidhar, Sathappan Muthiah, Kiyoshi Nakayama, Ratnesh Sharma, and Naren Ramakrishnan. 2019. Multivariate long-term state forecasting in cyber-physical systems: A sequence to sequence approach. In *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 543–552.

[24] Kevin P Murphy. 2022. *Probabilistic machine learning: an introduction*. MIT press.

[25] Mehtabhorn Obthong, Nongnuch Tantisantiwong, Watthanasak Jeamwatthanachai, and Gary Wills. 2020. A survey on machine learning for stock price prediction: Algorithms and techniques. (2020).

[26] Kyong Jo Oh and Ingoo Han. 2000. Using change-point detection to support artificial neural networks for interest rates forecasting. *Expert systems with applications* 19, 2 (2000), 105–115.

[27] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *International conference on machine learning*. Pmlr, 1310–1318.

[28] Gurnain Kaur Pasricha. 2006. Kalman filter and its economic applications. (2006).

[29] Pranav Patel, Eamonn Keogh, Jessica Lin, and Stefano Lonardi. 2002. Mining motifs in massive time series databases. In *2002 IEEE International Conference on Data Mining, 2002. Proceedings*. IEEE, 370–377.

[30] Debprakash Patnaik, Manish Marwah, Ratnesh K Sharma, and Naren Ramakrishnan. 2011. Temporal data mining approaches for sustainable chiller management in data centers. *ACM Transactions on Intelligent Systems and Technology (TIST)* 2, 4 (2011), 1–29.

[31] Fotios Petropoulos, Daniele Apiletti, Vassilios Assimakopoulos, Mohamed Zied Babai, Devon K Barrow, Souhaib Ben Taieb, Christoph Bergmeir, Ricardo J Bessa, Jakub Bijak, John E Boylan, et al. 2022. Forecasting: theory and practice. *International Journal of Forecasting* 38, 3 (2022), 705–871.

[32] Anna Rakitianskaia and Andries Engelbrecht. 2015. Measuring saturation in neural networks. In *2015 IEEE symposium series on computational intelligence*. IEEE, 1423–1430.

[33] Srinath Ravikumar and Prasad Saraf. 2020. Prediction of stock prices using machine learning (regression, classification) Algorithms. In *2020 International Conference for Emerging Technology (INCET)*. IEEE, 1–5.

[34] Claudio Filipi Gonçalves Dos Santos and João Paulo Papa. 2022. Avoiding overfitting: A survey on regularization methods for convolutional neural networks. *ACM Computing Surveys (CSUR)* 54, 10s (2022), 1–25.

[35] Omer Berat Sezer, Mehmet Ugur Gudelek, and Ahmet Murat Ozbayoglu. 2020. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied soft computing* 90 (2020), 106181.

[36] Fei Tan, Xiurui Hou, Jie Zhang, Zhi Wei, and Zhenyu Yan. 2018. A deep learning approach to competing risks representation in peer-to-peer lending. *IEEE transactions on neural networks and learning systems* 30, 5 (2018), 1565–1574.

[37] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, Eftychios Protopapadakis, et al. 2018. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience* 2018 (2018).

[38] Robert L Winkler. 1973. Bayesian models for forecasting future security prices. *Journal of Financial and Quantitative Analysis* 8, 3 (1973), 387–405.

[39] Shengzhe Xu, Manish Marwah, Martin Arlitt, and Naren Ramakrishnan. 2021. Stan: Synthetic network traffic generation with generative neural models. In *Deployable Machine Learning for Security Defense: Second International Workshop, MLHat 2021, Virtual Event, August 15, 2021, Proceedings 2*. Springer, 3–29.

[40] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. 2019. A review of recurrent neural networks: LSTM cells and network architectures. *Neural computation* 31, 7 (2019), 1235–1270.

[41] Yang Zhao, Jianping Li, and Lean Yu. 2017. A deep learning ensemble approach for crude oil price forecasting. *Energy Economics* 66 (2017), 9–16.