WATTNet: Learning to Trade FX via Hierarchical Spatio-Temporal Representation of Highly Multivariate Time Series

Michael Poli[1,2*], **Jinkyoo Park**[1,*] and **Ilija Ilievski**[2]
[1]Department of Industrial & Systems Engineering, KAIST, Daejeon, South Korea
[2]Neuri Pte Ltd, Singapore, Singapore
{poli_m, jinkyoo.park}@kaist.ac.kr, ilija139@neuri.ai

Abstract

Finance is a particularly challenging application area for deep learning models due to low noise-to-signal ratio, non-stationarity, and partial observability. *Non-deliverable-forwards* (NDF), a derivatives contract used in foreign exchange (FX) trading, presents additional difficulty in the form of long-term planning required for an effective selection of start and end date of the contract. In this work, we focus on tackling the problem of NDF position length selection by leveraging high-dimensional sequential data consisting of spot rates, technical indicators and expert tenor patterns. To this end, we curate, analyze and release a dataset from the Depository Trust & Clearing Corporation (DTCC) NDF data that includes a comprehensive list of NDF volumes and daily spot rates for 64 FX pairs. We introduce *W*aveA*TT*ention*Net* (WATTNet), a novel temporal convolution (TCN) model for spatio-temporal modeling of highly multivariate time series, and validate it across NDF markets with varying degrees of dissimilarity between the training and test periods in terms of volatility and general market regimes. The proposed method achieves a significant positive return on investment (ROI) in all NDF markets under analysis, outperforming recurrent and classical baselines by a wide margin. Finally, we propose two orthogonal interpretability approaches to verify noise robustness and detect the driving factors of the learned tenor selection strategy.

1 Introduction

Following recent trends of successful AI adoption, the financial world has seen a significant surge of attempts at leveraging deep learning and reinforcement learning techniques across various application areas. Slowing down progress in this field are the particular properties of financial data: low signal-to-noise ratio [Guhr and Kälber, 2003], partial observability, and irregular sampling. Furthermore, AI breakthroughs in finance often go unpublished due to monetary incentives. Additional challenges are caused by the scarcity of datasets available, which are often limited in scope, difficult to acquire or for some application areas missing altogether.

As an attempt to alleviate some of these concerns, we release both a curated dataset and a novel model for foreign exchange (FX) futures trading. We focus our attention on a particular class of FX trading methods, *non-deliverable-forward* (NDF) contracts, which constitute an important open problem in finance and can serve as a challenging benchmark for supervised or reinforcement learning models. We formulate the learning problem as an optimal selection problem in which the model is tasked with selecting the end date of the forward contract (tenor) from a rich input containing past human trade patterns as well as spot rates and technical indicators. In particular, tenor selection is cast into a *direct imitation learning* [Judah, Fern, and Dietterich, 2012] framework, where the model learns policy directly from a set of expert execution trajectories without receiving a reward signal from the environment. The demonstrations are derived in a greedy fashion from spot rate data and the resulting input-output tuple is utilized to perform standard supervised learning.

A key difference of our approach compared to existing FX trading algorithms lies in the type of data relied upon for learning, which includes expert tenor patterns in addition to standard technical indicators. Such patterns are extracted from a large dataset containing trades from competitive market players assumed to be informed about market state and to act rationally in order to achieve higher returns. Leveraging this additional information allows the models to differentiate between profitable and non-profitable market conditions with improved accuracy, ultimately leading to higher returns.

Fundamentally important for finance are models capable of capturing inter and intra-dependencies in highly multivariate time series. Many, if not most, of such interaction terms are nonlinear and thus challenging to analyze with standard statistical approaches. A direct consequence has been the newfound popularity of data-driven models for financial forecasting tasks, in particular recurrent neural networks (RNN) and their variants. Recurrent models, while offering an intuitive approach to time series modeling, lack an explicit module to capture inter-dependencies and perform relational reasoning [Santoro et al., 2018]. A different approach to time series modeling relies on temporal convolutions (TCN) as its fundamental computational block. Particularly successful in this

[*]Contact authors

area of research is WaveNet [Oord et al., 2016], originally developed as a generative model for speech data. However, vanilla WaveNet and its derivative models are primarily designed to handle univariate time series and thus are ill-suited for highly multivariate financial time series. To bridge this gap, we introduce a new TCN model called *WaveATTention-Net* (WATTNet) that incorporates computationally efficient dilated convolutions for temporal learning of autoregressive effects, and self-attention modules to learn spatial, inter-time series interaction terms.

We summarize our main contributions as follows:

- We curate, analyze, and release a new dataset containing spot rates for 64 FX currencies, along with technical indicators and hourly frequency NDF contract trade data spanning the period from 2013 to 2019. Several models, including classical baselines (Momentum-1, Momentum-90) and recurrent baselines (GRUs, LSTMs) are evaluated against expert benchhmarks obtained from NDF data.
- We introduce *WATTNet*, a novel *temporal convolution* (TCN) architecture for spatio-temporal modeling. WATTNet is designed to extend WaveNet models to settings with highly multivariate time series data.
- We provide two orthogonal approaches to evaluate noise robustness and explain driving factors of the learned trading strategy, along with examples to highlight their efficacy.

2 Background and Related Work

Foreign Exchanges Trading in *forex* (FX) markets is generally done via *spot exchanges* or *forward exchanges*, where *spot rate* indicates the present expected buying rate. The *spot market* can be volatile and is affected by news cycles, speculation, and underlying market dynamics. On the other hand, forward exchanges contain a long-term planning component: two parties fix a binding amount and date of exchange and the profits are calculated by comparing currency rates at the start date and fix date. The difference between start date and fix date is commonly referred to as *tenor*.

Non-Deliverable-Forward An NDF operates similarly to *forward exchange* contracts and exists as a replacement to forward FX trades in closed markets. NDF markets are *over-the-counter*, meaning they operate directly between involved parties without supervision, and are generally more volatile due to limited market-depth. In NDF trades the parties agree on notional amounts of primary and secondary currency (e.g. dollar USD and korean won KRW) which define the *forward rate*. The currency amounts are not exchanged at the end of the contract: instead, NDF trades are cash-settled in USD, and the cash flow is computed as $R_{t,a} = (x_{t+a} - x_t)v_t$ where x_t is the spot rate at time t, x_{t+a} the spot rate at time $t + a$, a is the tenor and v is the notional amount.

2.1 Related Work

Deep Learning for FX trading Earlier attempts at utilizing the expressivity of neural networks in *forex* (FX) trading [Chan and Teong, 1995] predict technical indicators via shallow fully-connected neural networks. More recently [Czekalski, Niezabitowski, and Styblinski, 2015; Galeshchuk and Mukherjee, 2017; Petropoulos et al., 2017; Pathberiya, Tilakaratne, and Hansen, 2017] have leveraged various deep learning modeling techniques. However, these approaches focus on regular forex markets and short-term predictions and rely only on spot rates and technical indicators as informative features. Incorporating additional sources of data has been explored in [Nassirtoussi et al., 2015; Vargas, De Lima, and Evsukoff, 2017; Hu et al., 2018], particularly text extracted from financial news articles.

While the literature has no shortage of works in which reinforcement learning is applied to portfolio management [Yu et al., 2019], the FX markets remain comparatively unexplored. [Carapuço, Neves, and Horta, 2018] develops a short-term spot trading system based on reinforcement learning and obtains positive ROI in the EURUSD market. [Sornmayura, 2019] offers an analysis of deep Q-learning (DQN) performance on two FX instruments. We are not aware of any published work where deep learning or reinforcement systems are introduced to tackle FX trading in an NDF setting.

Spatio temporal modeling SNAIL [Mishra et al., 2017] obtains improvements over vanilla WaveNet [Oord et al., 2016] by adding a temporal attention layer between dilated convolutions. However, both vanilla WaveNet and SNAIL are originally designed to process univariate time series data and are thus unable to learn interaction terms between time series. ConvLSTM [Xingjian et al., 2015] introduce a convolution operation inside the LSTM cell to capture spatiotemporal information. A weakness of ConvLSTMs and similar approaches [Lai et al., 2018] is given by the prior assumption of structure in the spatial domain where features closer together are prioritized by the convolution operation, as is the case for example with video data. In general applications, the time series are arbitrarily concatenated as input data and locality assumptions do not hold. A more recent approach to spatio-temporal modeling based on Graph Neural Networks (GNNs) is Graph WaveNet [Wu et al., 2019] which utilizes dilated convolutions in the temporal axis and proposes a data–driven approach to estimate the adjacency matrix when unavailable.

3 NDF Tenor Selection

We begin by a formal description of the NDF tenor selection task, then discuss NDF dataset construction.

Notation A multivariate time series of dimension M and length T is indicated as $\{\mathbf{X}\}$. We use $\{\mathbf{x}_i\}$ for individual time series indexed by i. $x_{i,t}$ selects a scalar element of time series i at time index t . In particular, we indicate a slice across all time series at time t with $\{x_{1,t} \dots x_{M,t}\}$. Whenever the operations on $\{\mathbf{X}\}$ are batched, we add superscript j for single samples in the batch: $\{\mathbf{X}^j\}$. With batch size N, the resulting tensor $\{\mathbf{X}^1\} \dots \{\mathbf{X}^N\}$ has dimensions $N \times T \times M$. We define the set of admissible tenor choices as $\mathcal{A} := [1, \dots, 90]$.

Problem statement The NDF tenor selection task involves, for each trading day t in a period $\mathcal{T}$, finding a maximally profitable position for the NDF market under scrutiny, namely

$$a_t^* = \arg\max_{a \in \mathcal{A}} (x_{t+a} - x_t) \tag{1}$$

where a represents the tenor, or position length in days, $\mathcal{A}$ the set of admissible tenor choices and x_t the spot rate for the corresponding FX instrument. Longer tenors are generally more profitable at the expense of a higher volatility, commonly referred to as *risk-premia*. A successful trading agent thus has to find a difficult balance between risky, high return and safer, low return actions. We propose a solution that incorporates expert human agent knowledge into the system, distilled from historical NDF trade records.

3.1 NDF Dataset

The NDF dataset is utilized in several ways: historical data provides the extraction of valuable benchmarks based on expert trader actions. These benchmarks, named *Expert* and *Expert oracle*, are introduced to further contextualize the experimental performance.

Dataset description The NDF trade records have been collected from the The Depository Trust & Clearing Corporation (DTCC) database, containing historical records for 6 major NDF markets: USDCNY, USDIDR, USDINR, USDKRW, USDTWD. These records include start and end dates of each NDF contract, along with currency amounts for the period spanning 2013 to 2019. For each NDF market under consideration and each admissible tenor $a \in \mathcal{A}$, we generate a time series of volumes $v_{t,a}$ which includes a summation of the currency amounts over all NDF records at a specific day t. Since $|\mathcal{A}| = 90$, each NDF pair contributes with a 90-dimensional multivariate volume time series to the input, containing total currency amount $v_{t,a}$ at day t for each of the 90 tenors. This high–dimensionality further emphasizes the need for a model capable of processing and aggregating information across highly multivariate time series.

Expert benchmarks We refer to *Expert* as a trading agent that chooses tenors corresponding to maximum volumes $a_t = \arg\max_{a \in \mathcal{A}} v_{t,a}$. In addition to *Expert* we obtain a fictitious agent based on NDF records which is assumed to have partial future knowledge of the market dynamics, which we refer to as *Expert oracle*. *Expert oracle* is a filtered version of *Expert*: at each trading day t it selects the shortest tenor with positive return:

$$a_t = \arg\min_{a \in \mathcal{A}} \{a | (x_{t+a} - x_t) > 0\} \tag{2}$$

In particular, *Expert oracle* is designed to select shortest tenors to avoid a perfect accuracy exploitation of risk-premia which would set an unrealistic benchmark for any model. The choice of *Expert* and *Expert oracle* has been carried out to contextualize results against typical market players, whose positions are derived directly from NDF historic dataset. Both *Expert* and *Expert oracle* are used as a benchmark during the evaluation phase.

Auxiliary features In order to learn how to effectively choose NDF tenors, we augment the volume features described above with technical indicators and spot rates for 64 major and minor FX currency pairs. Daily FX spot rates serve as contextual market information and provide a frame of reference that aids the model in identifying profitable states. Raw financial market data is often augmented with handcrafted features to help combat noise and non-stationarity [Ntakaris et al., 2019]. To this end, we choose: *simple moving average* (SMA), *exponential moving average* (EMA), *moving average convergence divergence* (MACD), *rolling standard deviation* (RSD), *Bollinger bands* (BB), *ARIMA 1-day spot rate forecast*.

3.2 The Imitation Problem

Selecting a profitable tenor is challenging since it requires the model to balance between short tenors with smaller returns or long, risky tenors with a potentially greater return. One approach to training a tenor selection model is performing imitation learning on *Expert* or *Expert oracle* labels. Both have advantages and disadvantages; training on *Expert* allows for daily online training and thus reduces the need for the model to extrapolate to periods further into the future. This aspect can be particularly beneficial for turbulent markets that display frequent regime switches. *Expert oracle* labels, on the other hand, require information from up to N days in the future, with N being the maximum allowed tenor, since positive return filtering can only be performed by leveraging spot rate data. *Expert oracle* labels can be advantageous since they teach the model to be *risk-averse*; however, both approaches potentially include unwanted human-bias in the learned strategy. We propose an alternative approach in which the model learns from optimal greedy tenor labels obtained directly from market data. More specifically, we extract optimal tenor label a_t^* defined in (1) at time t from raw data, effectively turning future spot rate information into training labels for the model. As a result, the knowledge of expert NDF traders is not injected into the labels, but is instead leveraged by the model through the input NDF volume features $v_{t,a}$.

4 The WATTNet Model

Spatio-temporal modeling with WATTNet *WaveATTentionNet* (WATTNet) is a novel model designed for highly multivariate time series inputs. WATTNet includes *temporal modules*, tasked with independently aggregating information across time steps of univariate time series $\{\mathbf{x}\} \in \{\mathbf{X}\}$ and *spatial modules* which aggregate features across slices of all time series at a specific time t $\{x_{1,t}, \ldots x_{M,t}\}$. Temporal and spatial modules are alternated and allow for learning a hierarchical spatio-temporal representation. As is customary, we refer to the *width* of multivariate timeseries as the *spatial* dimension. An overview of the model is given in Figure 1.

Temporal learning Temporal learning is achieved by applying temporal dilated convolutions (TCN) to univariate time series $\{\mathbf{x}\} \in \{\mathbf{X}\}$. Given a convolution with kernel size k and dilation coefficient d, we compute the output at time t of a dilated convolution of $\{\mathbf{x}\}$ as $z_t = f(\sum_{i=1}^{k} w_i * x_{t-i*d})$ where w_i is the i^{th} weight of the convolutional kernel and f a non–linear activation function. Each univariate time series has access to its own set of convolutional weights $\mathbf{w}$ as temporal convolution operations are carried on independently.

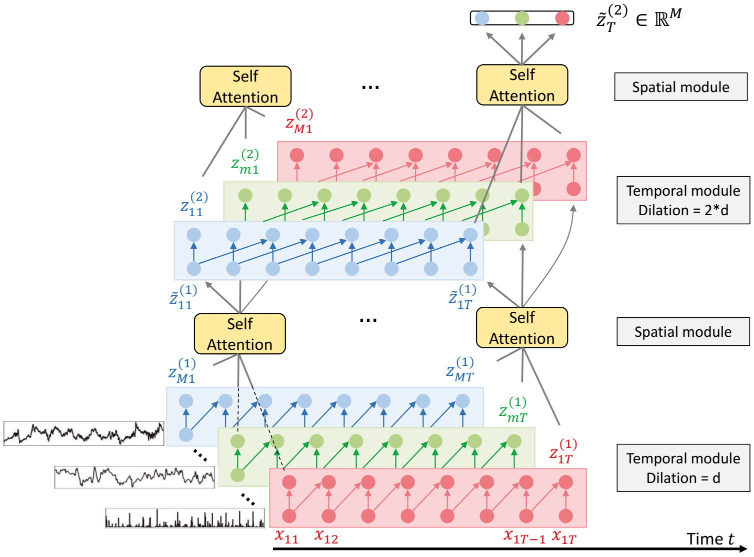

Figure 1: WATTNet overview: dilated TCNs are independently applied to each univariate input time series. A single dot-product attention head subsequently aggregates information across slices $\{z_{1,t} \dots z_{M,t}\}$ and the result $\{\tilde{\mathbf{Z}}\}$ is passed to the next WATTBlock or used directly.

We note that independence between convolutions is necessary to provide the model with enough flexibility to treat time series with different characteristics. The outputs of the TCN operation are then concatenated as to form a multivariate latent time series $\{\mathbf{Z}\}$. In particular, WATTNet includes gated convolutions, a standard architectural component for sequential data. Two dilated TCNs are applied to $\{\mathbf{X}\}$ and the results $\{\mathbf{Z}\}_\alpha$, $\{\mathbf{Z}\}_\beta$ are passed to non-linear activation functions and then multiplied element-wise:

$$\{\mathbf{Z}\} = \sigma(\{\mathbf{Z}\}_\alpha) \odot \tanh(\{\mathbf{Z}\}_\beta) \tag{3}$$

where σ indicates a sigmoid activation. The output $\{\mathbf{Z}\}$ is then fed into a spatial learning module after which the process repeats for a number of times depending on WATTNet's layer depth.

Spatial learning A single-head scaled-dot product attention mechanism [Vaswani et al., 2017] is placed between dilated TCN layers and allows the model to exchange information across different input time series at a specific time slice. We compute key $\mathbf{K}_t$, query $\mathbf{Q}_t$ and value $\mathbf{V}_t$ by considering a slice $\{z_{1,t} \dots z_{M,t}\} \in \{\mathbf{Z}\}$ of latent time series at time t as the input of learnable linear transformation of the type $\psi(\{z_{1,t} \dots z_{M,t}\})$ with weights $\mathbf{W}_k$, $\mathbf{W}_q$ and $\mathbf{W}_v$. The resulting matrices $\mathbf{K}_t$, $\mathbf{Q}_t$, and $\mathbf{V}_t$ are then used in the standard scaled-dot product attention to return M weighted averages of values $\mathbf{V}$:

$$\{\tilde{z}_{1,t} \dots \tilde{z}_{M,t}\} = \text{softmax}(\frac{\mathbf{Q}_t \mathbf{K}_t^T}{\sqrt{d_k}})\mathbf{V}_t \tag{4}$$

where d_k is a scaling factor given by the second dimension of $\mathbf{K}$. The process is repeated for latent feature slices $\{z_{1,t} \dots z_{M,t}\}$, $t = 1, \dots T$ and the results are concatenated into $\{\tilde{\mathbf{Z}}\}$, a spatio-temporal latent representation of input data $\{\mathbf{X}\}$. Weights $\mathbf{W}_k$, $\mathbf{W}_q$ and $\mathbf{W}_v$ are shared across the entire sequence length T, allowing the attention head to capture time-invariant features that incorporate information from multiple time series. Output $\{\tilde{\mathbf{Z}}\}$ can be used directly for different tasks to perform decision–making conditioned on multivariate time series data; alternatively, if the task at hand benefits from deeper models, $\{\tilde{\mathbf{Z}}\}$ can instead be passed to the following TCN layer to perform additional cycles of temporal and spatial learning.

Hierarchical representation A single temporal and spatial module constitute a full WATTNet layer of computation and is referred to as *WATTBlock*. WATTBlocks can be stacked, in which case output $\{\tilde{\mathbf{Z}}\}$ becomes a hierarchical spatio-temporal representation of $\{\mathbf{X}\}$. As is the case with other TCN-based models, the dilation coefficient is doubled each temporal module as to provide an increasing receptive field which allows for a computationally inexpensive way to model long sequences. An additional benefit of the gradual dilation increase is the slow introduction of interaction terms between time series which include less lagged values for early WATTBlocks and more for later ones. We observe that a gradually increasing the size of this interaction window is key in learning a hierarchical representation of the data that strongly intertwines spatial and temporal causal effects.

Prediction model The model ϕ is connected to a fully-connected head which takes as input the latent representation $\{\tilde{\mathbf{Z}}\}$ of a length T input sequence $\{\mathbf{X}_{[t-T:t]}\}$ and outputs a probability distribution over tenor actions via softmax. The training procedure is then carried out by minimizing, for each trading day t, the cross–entropy loss of $\phi(\{\mathbf{X}_{[t-T:t]}\})$ and tenor labels a_t^*.

5 Experimental Results

The experimental evaluation covers the following 3 major NDF markets: *Chinese Yuan* (USDCNY), *Korean Won* (USDKRW), *Indonesian Rupiah* (USDIDR). The selection has been carried out to test the proposed method on markets with

	USDCNY			USDKRW			USDIDR		
Model	ROI	opt.acc	nn.acc	ROI	opt.acc.	nn.acc.	ROI	opt.acc.	nn.acc.
Optimal	759.8	100	100	844.2	100	100	1260.0	100	100
Expert (oracle)	77.6	21.6	100	139.4	13.7	100	152.0	3.6	100
Expert	0.0	1.4	47.8	12.7	1.9	49.9	230.0	0.4	67.0
Momentum-1	14.6	1.4	48.3	10.7	0.9	49.7	201.0	0.7	66.5
Momentum-90	4.9	6.7	54.1	31.9	2.6	56.2	**338.0**	**1.9**	69.2
GRU-I	26.7. ± 48.5	4.9 ± 0.7	54.7 ± 1.0	−98.7 ± 35.4	1.5 ± 0.5	52.1 ± 1.4	83.5 ± 33.1	0.6 ± 0.1	62.9 ± 2.6
LSTM-I	74.3 ± 37.3	3.7 ± 0.9	58.7 ± 2.2	74.6 ± 30.0	2.6 ± 0.6	56.0 ± 2.5	146.4 ± 40.4	1.1 ± 0.3	66.5 ± 1.1
WATTNet	**219.1** ± 25.5	**6.7** ± 0.7	**59.5** ± 1.6	**142.4** ± 16.9	**2.7** ± 0.2	**59.5** ± 1.0	280.2 ± 39.9	1.3 ± 0.3	**69.5** ± 0.9

Table 1: Test results in percentages (average and standard error). Best performance is indicated in bold.

Market	μ_{train}	σ_{train}	μ_{test}	σ_{test}
USDCNY	1.13	23.16	-0.30	23.29
USDIDR	2.20	50.62	0.35	33.57
USDKRW	0.05	47.12	-1.27	38.75

Table 2: Statistics of daily percent returns, reported in basis points.

different characteristics as shown in Table 2.

Classical baselines We include the following classical trading baselines:

- *Momentum-1*: 1-day lag of expert tenor actions. Effective in markets where monotonic behavior in the spot rate is frequent and persists for several trading periods.
- *Momentum-90*: best performing tenor from 90 days prior. Effective in markets with trends whose duration is longer compared to the maximum tenor.

Model hyperparameters GRU and LSTMs are used as additional baselines for tenor and referred to as GRU-I and LSTM-I. GRU-I and LSTM-I share the same structure, which has been optimized for the task at 2 recurrent layers with latent vector dimension of 512. The WATTNet model is composed of 8 WATTBlocks with the dilation coefficients for temporal learning set to 2, 4, 8, 16, 2, 4, 8, 16 for WATTBlock-1 to 8. All models are connected to a 2–layer fully-connected head of dimensions $512, 256, 90$ which transforms the spatio–temporal, latent representation of $\{\mathbf{X}\}$ produced by the model in its last layer and produces a probability distribution over tenor actions via softmax. We further augment $\mathcal{A}$ with a *no–buy* position, yielding a 91–dimensional set of available actions.

Training setup The models are implemented in PyTorch and trained using Adam [Kingma and Ba, 2015] with batch size 32 and a learning rate cosine decay schedule from $6e^{-4}$ down to $3e^{-4}$. To avoid overfitting uninformative noisy patterns in stale data input sequence length is set to 30 days. In addition, to enable a fair comparison and avoid additional overfitting we employ an early stopping scheme based on training loss that is motivated by different convergence times of different models. We use a static testing approach with a long period of 446 out-of-sample trading days to test robustness of the learned trading strategy under turbulent market conditions and a wider distributional shift between in-sample and out-of-sample data. The training dataset ranges from 2013 to 2017, whereas test data spans 2017 to 2019.

Metrics The following metrics are used to benchmark the performance of trading models and baselines:

- *Return on investment* (ROI): given a tenor action a at time t and spot rate value x_t, the percent non–risk adjusted ROI is calculated as $\text{ROI}_t = 100\left(\frac{x_{t+a}-x_t}{x_t}\right)$
- *Optimal accuracy*: accuracy of model outputs $\hat{a}_t$ versus *optimal* tenor labels a^*_t.
- *Non-negative return accuracy*: accuracy of model outputs $\hat{a}_t$ compared to tenor actions with positive or zero return. At time t, there are generally multiple tenor actions with non-negative return, thus rendering *non-negative accuracy* a less strict metric compared to *optimal accuracy*. It should be noted that it is possible for a high ROI trading model to show poor optimal accuracy but competitive positive return accuracy since non-negative accuracy also captures positive ROI strategies that differ from optimal tenor labels.

Discussion We characterize the 3 NDF markets under evaluation based on mean μ and standard deviation σ of their 1-day returns and performance of classical baselines. Mean-variance statistics given in Table 2 show ”easier” markets with similar volatility in training and test periods as well as markets that appear more erratic and thus challenging to trade profitably in. From Table 1, we determine USDCNY and USDKRW to be challenging for *Momentum* and recurrent baselines, in addition to being barely profitable for expert traders. GRU-I is unable to consistently get positive ROI and we suspect this is caused by its insufficient ability to exchange information between time series. LSTM-I, on the other hand, fares better by leveraging its memory module to perform rudimentary spatial learning [Santoro et al., 2018].

6 Explainability

Model explainability is particularly important in critical decision–making applications, such as algorithmic trading. Understanding driving factors behind a trading decision is necessary to properly assess the risks involved. We tackle this issue by proposing two orthogonal approaches for evaluating the tenor selection strategy in terms of noise robustness and driving factors.

6.1 Feature Importance By Input Gradients

Each input feature carries a specific meaning which can be leveraged by domain experts to confirm whether the model

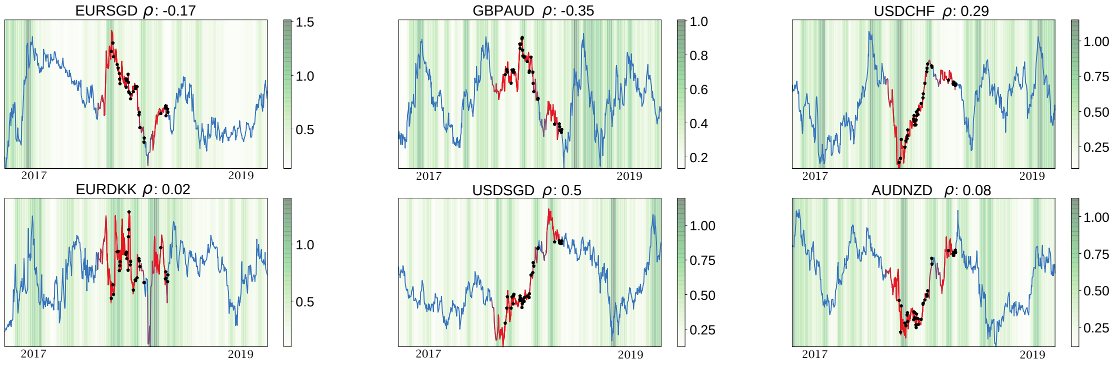

Figure 2: Highest impact spot rate features behind 90 day tenor USDCNY actions. Background color shows 20 day rolling standard deviation.

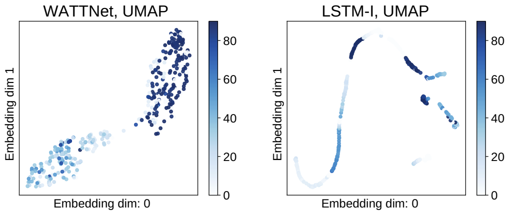

Figure 3: UMAP embedding of model latents. The points are labeled according to the final tenor output of the model.

outputs actions consistent with market dynamics. We pinpoint the driving factors of different tenor actions by sorting the input features by their *per–class input gradient* magnitude [Baehrens et al., 2010]. Input gradients allow the identification of features with largest impact on specific model decisions. The proposed approach is validated by an analysis of the driving factors for long tenor WATTNet positions in the USDCNY NDF market.

Discussion Fig. 2 shows historical data of the 6 highest impact input features for learned tenors of 90 days. Blue indicates the spot rate itself, whereas the shaded green background corresponds to the 20 day rolling standard deviation, highlighting regions of low and high market volatility. We compute the Pearson's correlation coefficient ρ between the USDCNY spot rate, the NDF market under consideration, and the time series of each feature. In red, the plot shows input segments that are mapped by WATTNet to long 90–day tenor actions. The black dots indicate the exact day in which a long position was initiated by the model. The model is confident to choose long positions when currencies that are positively correlated with USDCNY, such as USDSGD, undergo periods of growth. Similarly, decreasing trend periods for spot rates negatively correlated with USDCNY are preferred. Indeed, the degree to which such trends affect the model is directly reflected in ρ: USDCHF, still positively correlated with USDCNY, shows a less decisive positive trend, with additional ups and downs. Moreover, the model learns to favor trading periods with low volatility. To summarize, WATTNet correctly identifies trends in the driving features and utilizes this contextual market landscape information to determine profitable long positions. The analysis can be extended by domain experts to additional input features, such as technical indicators or past tenor actions, boosting confidence in model decisions.

6.2 Latent Space Representation

Desired properties of the learned trading strategy are *input coherence* and *robustness*. Input coherence is characterized by similar tenor outputs for similar states. Robustness, on the other hand, is concerned with how much noise perturbation is required to cause a tenor switch from a certain state. We perform a visual inspection of these properties via a *uniform manifold approximation and projection* (UMAP) which excels at capturing both local and global structure of the high-dimensional data [McInnes, Healy, and Melville, 2018]. Fig. 3 visualizes latent representation of the last layer for each model, embedded into two-dimensional space. More specifically, we apply UMAP to the spatio–temporal representation preceding the fully–connected head defined in Sec. 5. While Fig. 3 shows latent representations learned for the USDCNY NDF market, the discussion has been observed to be general.

Discussion WATTNet learns a coherent representation that smoothly interpolates between tenor actions and covers a larger volume of the embedding space. On the other hand, LSTM-I and GRU-I learn a thin lower–dimensional latent representation that mixes tenor labels abruptly. For recurrent models, small noise perturbations can then cause wide jumps in tenor actions and lower the performance, partially explaining the result volatility of GRU-I and LSTM-I observed in Table 1.

7 Conclusion

We introduced a challenging imitation learning problem, tenor selection for *non-deliverable-forward* (NDF) contracts. With the goal of promoting further research in this direction, we constructed and released a comprehensive NDF dataset and designed *WaveATTentioNet* (WATTNet) a novel deep learning model for spatio-temporal data which outperforms expert benchmarks and traditional baselines across several NDF markets. Finally, we employed two explainability techniques to determine driving factors and noise robustness of the learned tenor strategy.

References

[Baehrens et al., 2010] Baehrens, D.; Schroeter, T.; Harmeling, S.; Kawanabe, M.; Hansen, K.; and MÃžller, K.-R. 2010. How to explain individual classification decisions. *Journal of Machine Learning Research* 11(Jun):1803–1831.

[Carapuço, Neves, and Horta, 2018] Carapuço, J.; Neves, R.; and Horta, N. 2018. Reinforcement learning applied to forex trading. *Applied Soft Computing* 73:783–794.

[Chan and Teong, 1995] Chan, K. C., and Teong, F. K. 1995. Enhancing technical analysis in the forex market using neural networks. In *Proceedings of ICNN'95-International Conference on Neural Networks*, volume 2, 1023–1027. IEEE.

[Czekalski, Niezabitowski, and Styblinski, 2015] Czekalski, P.; Niezabitowski, M.; and Styblinski, R. 2015. Ann for forex forecasting and trading. In *2015 20th International Conference on Control Systems and Computer Science*, 322–328. IEEE.

[Galeshchuk and Mukherjee, 2017] Galeshchuk, S., and Mukherjee, S. 2017. Deep networks for predicting direction of change in foreign exchange rates. *Intelligent Systems in Accounting, Finance and Management* 24(4):100–110.

[Guhr and Kälber, 2003] Guhr, T., and Kälber, B. 2003. A new method to estimate the noise in financial correlation matrices. *Journal of Physics A: Mathematical and General* 36(12):3009.

[Hu et al., 2018] Hu, Z.; Liu, W.; Bian, J.; Liu, X.; and Liu, T.-Y. 2018. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 261–269. ACM.

[Judah, Fern, and Dietterich, 2012] Judah, K.; Fern, A. P.; and Dietterich, T. G. 2012. Active imitation learning via reduction to iid active learning. In *2012 AAAI Fall Symposium Series*.

[Kingma and Ba, 2015] Kingma, D. P., and Ba, J. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

[Lai et al., 2018] Lai, G.; Chang, W.-C.; Yang, Y.; and Liu, H. 2018. Modeling long-and short-term temporal patterns with deep neural networks. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 95–104. ACM.

[McInnes, Healy, and Melville, 2018] McInnes, L.; Healy, J.; and Melville, J. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

[Mishra et al., 2017] Mishra, N.; Rohaninejad, M.; Chen, X.; and Abbeel, P. 2017. A simple neural attentive meta-learner. *arXiv preprint arXiv:1707.03141*.

[Nassirtoussi et al., 2015] Nassirtoussi, A. K.; Aghabozorgi, S.; Wah, T. Y.; and Ngo, D. C. L. 2015. Text mining of news-headlines for forex market prediction: A multi-layer dimension reduction algorithm with semantics and sentiment. *Expert Systems with Applications* 42(1):306–324.

[Ntakaris et al., 2019] Ntakaris, A.; Mirone, G.; Kanniainen, J.; Gabbouj, M.; and Iosifidis, A. 2019. Feature engineering for mid-price prediction with deep learning. *IEEE Access* 7:82390–82412.

[Oord et al., 2016] Oord, A. v. d.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; and Kavukcuoglu, K. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*.

[Pathberiya, Tilakaratne, and Hansen, 2017] Pathberiya, H. A.; Tilakaratne, C. D.; and Hansen, L. 2017. An intelligent system for forex trading: Hybrid ann with garch and intrinsic mode functions. In *2017 Intelligent Systems Conference (IntelliSys)*, 436–445. IEEE.

[Petropoulos et al., 2017] Petropoulos, A.; Chatzis, S. P.; Siakoulis, V.; and Vlachogiannakis, N. 2017. A stacked generalization system for automated forex portfolio trading. *Expert Systems with Applications* 90:290–302.

[Santoro et al., 2018] Santoro, A.; Faulkner, R.; Raposo, D.; Rae, J.; Chrzanowski, M.; Weber, T.; Wierstra, D.; Vinyals, O.; Pascanu, R.; and Lillicrap, T. 2018. Relational recurrent neural networks. In *Advances in Neural Information Processing Systems*, 7299–7310.

[Sornmayura, 2019] Sornmayura, S. 2019. Robust forex trading with deep q network (dqn). *ABAC Journal* 39(1).

[Vargas, De Lima, and Evsukoff, 2017] Vargas, M. R.; De Lima, B. S.; and Evsukoff, A. G. 2017. Deep learning for stock market prediction from financial news articles. In *2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, 60–65. IEEE.

[Vaswani et al., 2017] Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

[Wu et al., 2019] Wu, Z.; Pan, S.; Long, G.; Jiang, J.; and Zhang, C. 2019. Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, 1907–1913. International Joint Conferences on Artificial Intelligence Organization.

[Xingjian et al., 2015] Xingjian, S.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.-K.; and Woo, W.-c. 2015. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, 802–810.

[Yu et al., 2019] Yu, P.; Lee, J. S.; Kulyatin, I.; Shi, Z.; and Dasgupta, S. 2019. Model-based deep reinforcement learning for dynamic portfolio optimization. *arXiv preprint arXiv:1901.08740*.