Machine Learning for Realised Volatility Forecasting

Preprint *in* SSRN Electronic Journal · October 2020

DOI: 10.2139/ssrn.3707796

2 authors:

Eghbal Rahimikia
The University of Manchester
7 PUBLICATIONS **70** CITATIONS
SEE PROFILE

Ser-Huang Poon
The University of Manchester
139 PUBLICATIONS **5,265** CITATIONS
SEE PROFILE

Machine Learning for Realised Volatility Forecasting

Eghbal Rahimikia and Ser-Huang Poon*

First version: October 12, 2020 • This revision: November 11, 2023

Abstract

We examine the predictive power of machine learning (ML) models for forecasting realised volatility (RV) using different data sets tested in the literature, viz., variables from the HAR models, limit order book (LOB), and news sentiments. With 3.7 million ML models trained and robustness checked, the high dimensional ML models (viz. 147 data features for up to 21 lags) significantly outperformed HAR on 90% of the out-of-sample period when the actual volatility is not extreme. SHAP values reveal that mid prices, mean bids, and mean asks are the most useful predictive variables. The ML models also better than HAR at capturing forecast dynamics as they evolve through time.

Keywords: Realised Volatility Forecasting, Machine Learning, Big Data, Long Short-Term Memory, Heterogeneous AutoRegressive Models, Explainable AI.

JEL: C22, C45, C51, C53, C55, C58

*Eghbal Rahimikia (corresponding author) (eghbal.rahimikia@manchester.ac.uk) and Ser-Huang Poon (ser-huang.poon@manchester.ac.uk) are at the University of Manchester, Alliance Manchester Business School, UK. In particular, we would like to extend our thanks to Robert Engle, Francis X. Diebold, and Bill McDonald for their hints and tips. We are also grateful to the participants and discussants of the British Accounting and Finance Association (BAFA) Annual Conference, 37th International Conference of the French Finance Association (AFFI), 8th Annual MMF PhD Conference, Durham University Business School, 7th International Young Finance Scholar's Conference, Data Fest, Russia, Finance Research Day, Alliance Manchester Business School, and 10th International Conference of the Financial Engineering and Banking Society (FEBS). We also extend our gratitude to all anonymous referees for their exceptional feedback. All models are run on the computational shared facility of the University of Manchester. We must express our sincere appreciation to the IT services of the University of Manchester for their constant and continued support and for providing the computational infrastructures for this study. Last but not least, Our sincere thanks are due to the accounting and finance division at Alliance Manchester Business School for their financial support.

1 Introduction

Volatility forecasting plays a critical role in financial modelling and financial decision-making. This paper studies the effectiveness of machine learning (ML) models combined with a rich feature set in volatility forecasting for 23 NASDAQ stocks over a sample period from 27 July 2007 to 27 January 2022. To date, no study has tested ML models' volatility forecasting performance with such a rich dataset and long sample period. It is vital to consider ML for several reasons. First, the classical econometric models, without undergoing additional modifications, cannot handle a large number of input variables in big datasets, such as LOB and news stories, in a single model. Second, ML models are usually more efficient in capturing nonlinear relationships in high dimensions, a task that standard econometric models generally struggle to accomplish. In this study, we aim to explore the impact of a wide array of variable clusters originating from well-known HAR-family of models, high-frequency limit order book (LOB) data, and sentiment variables extracted from news stories on the predictive accuracy of realised volatility (RV) under different market conditions, achieved through the utilisation of machine learning models, notably long short-term memory (LSTM) models.

ML models have been used in various finance topics for some years. Chen et al. (2019) exploited the ML model in the estimation of the stochastic discount factor, Gu et al. (2020) showed superior performance of ML models for empirical asset pricing, Jiang et al. (2020) produced more accurate stock return predictions based on ML image analyses, and Gu et al. (2021) introduced auto-encoder asset pricing model and produced smaller out-of-sample pricing errors compared to classical leading factor models. A subgroup of ML models has been developed for special types of sequential data such as video, music, text, and, in our case, financial time series. Recurrent neural network (RNN) and its extension, long short-term memory (Hochreiter and Schmidhuber, 1997), are among the most widely used pioneering ML models in both academia and industry for this type of sequential data paving the way for more advanced ML models (see Vinyals et al. (2019) and Andrychowicz et al. (2020) for recent LSTM use cases). LSTM is currently the most successful type of RNN, and it is responsible for many state-of-the-art sequence modelling (Goldberg, 2016). By employing a chain structure, LSTM models can perform sequential processing and learn the long-term dependencies of time series.

The availability of high-frequency financial data makes RV a popular model-free and low measurement error proxy for actual volatility (Andersen et al., 2001; Barndorff-Nielsen and Shephard, 2001). With the popularity of RV, the heterogeneous autoregressive (HAR) model (Corsi, 2009) and its variations are commonly used in forecasting RV. The well-known variants include HAR-J (HAR with jumps) and CHAR (continuous HAR) (Andersen et al., 2007; Corsi and Reno, 2009), Patton and Sheppard (2015) SHAR (semivariance-HAR) that separates the

impact of negative and positive returns on the subsequent RV, and Bollerslev et al. (2016) HARQ model adjusting the forecasts for measurement error based on realised quarticity (RQ).

Within the realm of literature centred on ML models for RV forecasting, Hillebrand and Medeiros (2010), Fernandes et al. (2014), and Audrino and Knaus (2016) demonstrated that ML models exhibit similar or lower forecasting performance when compared to HAR-family of models. On the other hand, Bucci (2020); Christensen et al. (2022) reported improvement in RV forecasting using ML models. Apart from fitting ML models with a large comprehensive dataset, it is very important to fully explore the hyperparameters and tuning of ML models. This makes detailed and large-scale exploratory study of multiple ML models infeasible in a single study. In contrast to these previous ML studies that utilised a relatively small number of variables, our paper focused on leveraging a vast array of data features and testing over 3,689,200 variants of our ML model. This approach enables us to explore thoroughly if ML (with a wide and varied range of variables) outperforms HAR-family of models in RV forecasting.

The three variable sets, i.e., LOB, News sentiments and HAR, are commonly used in the literature for volatility forecasting though rarely all three at the same time. The high dimensionality of ML models allow us to test all these variables together in a single model. Using combinations of 147 LOB, News sentiments and HAR variables, this paper provides strong statistical evidence that ML models outperforms the HAR-family of models in RV forecasting according to MSE, QLIKE (Quasi-LIKE), and MDA (Mean Directional Accuracy) loss functions as well as reality check (RC). This considerable and statistically significant enhancement applies to 90% of out-of-sample daily forecasts, except when the actual RV reaches high levels. On high volatility days, the HAR-family of models generally outperform ML models. LOB data, in general, provides stronger volatility forecasting performance when compared to News Sentiment variables. One important discovery of our study is the importance of incorporating a wide range of input variables in a single model to improve RV forecasting performance. We noted that even a simple ML model with HAR variables can outperform HAR models in forecasting RV. As emphasised by Christensen et al. (2022), this could be due to the ability of ML models to capture nonlinear relationships in RV forecasting. However, the performance improvement becomes particularly pronounced when a rich set of predictors is included in the ML models.

Recently, there has been an increased attention in the fields of accounting and finance that deploys *Explainable AI* technique to evaluate ML models, see, for example, Bali et al. (2021); Erel et al. (2021); and Chronopoulos et al. (2023). By applying SHAP (SHapley Additive exPlanations), an *Explainable AI* technique, we find, among the 147 input variables, mid prices at all LOB levels, average bid, and average ask possess the most predictive power for fore-

casting RV. Performance variations over time are also evident. The News sentiment variables exhibited robust predictive power before 2018. Particularly noteworthy is HAR's strong forecasting performance during the disruptive period of COVID-19 in 2019 and the exceptionally high volatility period in early 2020. LOB variables dominated among the predictors for RV forecasts from 2020. This alternation in the importance of different predictors over time once again underscores the critical importance of the time-varying choice of optimal predictors for RV forecasting, which is feasible in ML models.

Finally, robustness checks confirmed that for normal volatility days in the out-of-sample period, the superiority of ML over HAR-family of models remained unchanged. In total, we trained and tested 3,689,200 LSTM models for individual stocks using different ML hyperparameters, i.e., the number of units and the number of epochs. For the 90% of the out-of-sample forecasting period, when the actual RV level is less extreme, the optimal number of units and epochs vary, but the overall ML structure is simpler. For the 10% of the out-of-sample forecasting period when the actual RV level is extreme, a more complex ML structure with a large number of units and epochs is needed.

The remaining of this paper is organised as follows: Section 2 gives a brief review of RV and HAR-family of models. Section 3 describes the data and provides variable definitions. Section 4 provides the background of ML; the RNN model in Subsection 4.1, the LSTM model in Subsection 4.2, the regularisation in Subsection 4.3, and the structure of our proposed ML models in Subsection 4.4. Section 5 presents the results of the primary experiments, Section 6 evaluates the predictive power of input variables using SHAP, and Section 7 performs a series of robustness checks. Finally, Section 8 concludes with a discussion of the findings from this study.

2 Realised Volatility and HAR-Family of Models

Suppose that P_t is the stock price process with the following dynamics:

$$d\log(P_t) = \mu_t dt + \sigma_t dw_t + J_t dq_t, \tag{1}$$

where μ_t is the drift (continuous function), σ_t is the volatility process (càdlàg function), J_t is the jump size, w_t is the standard Brownian motion, and q_t is a Poisson process. For time $t-1$ to t, the integrated variance is defined as follows:

$$IV_t = \int_{t-1}^{t} \sigma_s^2 ds. \tag{2}$$

This integrated variance is not observable; therefore, quadratic variation, QV is used to proxy RV as follows:

$$RV_t \equiv \sum_{i=1}^{M} r_{t,i}^2, \tag{3}$$

where M is the sampling frequency and $r_{t,i} \equiv \log(P_{t-1+i\delta}) - \log(P_{t-1+(i-1)\delta})$. For $\delta \to 0$, QV_t is a consistent estimator for IV_t (Barndorff-Nielsen and Shephard, 2002).

2.1 HAR-Family of Models

Introduced by Corsi (2009), the HAR-family of models, defined below, includes the most popular RV forecasting models:

$$RV_{t+1} = \beta_0 + \beta_1 RV_t + \beta_2 \overline{RV}_t^w + \beta_3 \overline{RV}_t^m + \epsilon_{t+1}, \tag{4}$$

where RV_{t+1} is the forecast of time $t+1$ RV, RV_t is daily RV at time t, $\overline{RV}_t^w$ is the daily average over the last week, and $\overline{RV}_t^m$ is the daily average over the last 21 days. Corsi (2009) showed that this easy-to-estimate linear model with a simple set of historical RVs produced remarkable forecasting performance. It was since the work of Corsi (2009) that the research on improving RV forecasting performance has gained momentum. Researchers expanded the basic HAR model with various information sets and high-frequency data to enhance RV forecasting performance.

Andersen et al. (2007) and Corsi and Reno (2009) analysed the impact of adding a jump component to the basic HAR model. Define the jump component at time t as $J_t = Max[RV_t - BPV_t, 0]$ with bipower variations:

$$BPV_t = \frac{\pi}{2} \sum_{i=1}^{M-1} |r_{t,i}||r_{t,i+1}|, \tag{5}$$

where M is the maximum value of sampling frequency and $r_{t,i}$ is the return at the day t, and sampling frequency i. BPV_t was found to persist and be useful for forecasting RV.

As a variation to Equation (4), the CHAR model, introduced by Corsi (2009), replaces the predictive variables with BPV_t as follow:

$$RV_{t+1} = \beta_0 + \beta_1 BPV_t + \beta_2 \overline{BPV}_t^w + \beta_3 \overline{BPV}_t^m + \epsilon_{t+1}, \tag{6}$$

where BPV_t, $\overline{BPV}_t^w$, and $\overline{BPV}_t^m$ are, respectively, the daily BPV, the daily average over the past week and the daily average over the past month at time t. Without the jump component, BPV in Equation (6) is better at capturing volatility persistence and long memory than RV in

Equation (4).

Patton and Sheppard (2015) proposed the SHAR model separating the impact of negative and positive intraday returns on the subsequent RV. In the SHAR model, the first lag of RV in the HAR model (Equation (4)) is replaced by a positive return RV_t^+, and a negative return RV_t^-, where $RV_t^+ = \sum_{i=1}^{M} r_{t,i(r>0)}^2$ and $RV_t^- = \sum_{i=1}^{M} r_{t,i(r<0)}^2$. The authors found RV_{t+1} is more strongly related to RV_t^- than RV_t^+ for the S&P 500 index and 105 individual stocks.

Bollerslev et al. (2016) studied the impact of measurement error on volatility forecasting. First, they defined the integrated quarticity, $IQ_t = \int_{t-1}^{t} \sigma_s^4 ds$ and its discrete time equivalent, realised quarticity $RQ_t \equiv (\frac{M}{3}) \sum_{i=1}^{M} r_{t,i}^4$. Next, they introduced the ARQ, HARQ, and HARQ-F models. The HARQ model is defined as follows:

$$RV_{t+1} = \beta_0 + \beta_1 RV_t + \beta_{1Q} RQ_t^{1/2} RV_t + \beta_2 \overline{RV}_t^w + \beta_3 \overline{RV}_t^m + \epsilon_{t+1}, \tag{7}$$

where $RQ^{1/2}$ is demeaned for easier interpretation. For $\beta_{1Q} < 0$, RV_t has a lower impact when the measurement error is larger and a higher impact when the measurement error is smaller. When $\overline{RV}_t^w = 0$ and $\overline{RV}_t^m = 0$, Equation (7) reduces to the ARQ model. Also, HARQ-F is defined by adding the daily average of RQ over the past week and past month to the HARQ model specification. Bollerslev et al. (2016) found the HARQ model to have better forecasting performance, producing more volatility persistency in normal times and quicker volatility mean reversion in erratic times for the S&P 500 index and 27 Dow Jones constituent stocks.

Finally, Rahimikia and Poon (2022) conducted an extensive comparison of all HAR-family of models and found the CHAR model to be the best-performing model for forecasting RV. Also, they introduced CHARx, an extension of the CHAR model, by adding a variety of variables extracted from LOB data and news stories. They concluded that adding just the past day's average LOB depth and news count to CHAR statistically improves the forecasting performance of high volatility days.[1]

3 Variables

In this study, 23 *NASDAQ* stocks with the highest liquidity from 27 July 2007 to 27 January 2022 were selected. RV is calculated according to Equation (3) using 5-minute stock returns. Table 1 provides the RV descriptive statistics for these 23 stocks. Subsection 3.1, Subsection 3.2,

[1] As highlighted by Andersen et al. (2007), considering logarithmic RV could serve as a viable approach for controlling RV extreme distributions. In alignment with the approach taken by Bollerslev et al. (2016) and consistent with the prevailing literature on the HAR-family of models, this study uses the raw RV formulation instead of log-RV.

Table 1: Descriptive statistics of realised volatility of 23 NASDAQ stocks

Ticker[a]	Min	Max	1st quantile	Median	3rd quantile	Mean	STD	Kurtosis	Skewness
AAPL	0.102	229.420	0.899	1.733	3.680	4.623	12.596	111.012	9.124
MSFT	0.067	216.181	0.829	1.449	2.814	3.237	8.125	194.004	11.275
INTC	0.030	318.697	1.103	1.873	3.577	4.299	11.628	294.963	13.982
CMCSA	0.004	237.387	0.910	1.632	3.320	3.821	9.697	192.169	11.462
QCOM	0.122	373.543	1.024	1.975	4.129	5.073	15.380	200.609	12.100
CSCO	0.047	343.946	0.886	1.561	3.028	4.115	13.160	212.453	12.258
EBAY	0.205	252.608	1.319	2.271	4.356	5.082	12.592	142.684	10.009
GILD	0.064	259.489	1.167	1.892	3.379	4.304	12.930	182.820	12.063
TXN	0.177	287.897	1.047	1.905	3.748	4.014	9.820	311.666	14.242
AMZN	0.065	547.030	1.305	2.336	4.808	6.200	19.359	242.205	12.735
SBUX	0.052	265.094	0.864	1.594	3.423	4.201	11.237	161.435	10.626
NVDA	0.159	1104.351	2.282	4.358	9.084	9.756	30.117	586.612	20.058
MU	0.292	484.388	3.570	6.246	11.912	12.818	25.734	89.141	7.960
AMAT	0.292	531.579	1.783	3.028	5.712	6.005	14.632	532.194	18.338
NTAP	0.119	462.821	1.503	2.587	5.154	6.289	18.008	201.510	11.934
ADBE	0.119	569.720	1.099	2.020	3.908	4.947	15.003	588.095	18.867
XLNX	0.229	265.374	1.296	2.363	4.787	5.005	11.941	194.718	11.764
AMGN	0.032	214.156	0.969	1.593	2.872	3.398	9.612	183.759	11.898
VOD	0.055	219.033	0.687	1.342	3.137	3.933	10.869	122.252	9.601
CTSH	0.189	485.894	0.984	1.764	4.161	5.288	15.757	325.214	14.287
KLAC	0.154	499.808	1.456	2.710	5.416	5.919	16.878	354.626	16.033
PCAR	0.039	389.930	1.157	2.162	4.633	5.125	12.108	313.338	13.010
ADSK	0.268	693.772	1.644	2.765	5.167	6.644	22.377	388.131	16.554

and Subsection 3.3 describe the three variable sets from, respectively, HAR-family, news stories, and LOB. The LOB variables for these 23 stocks are compiled using information extracted from *LOBSTER*. (See Rahimikia and Poon (2022) for the data cleaning procedures.) These are the three main variable sets that are commonly used in the literature for volatility forecasting. The fourth variable set is simply the amalgamation of all three variable sets above.

3.1 HAR-Family Variables

Table 2 lists the variables used in the HAR-family of models. The first column ('Description') contains the name of the variables. 'RV', 'BPV' and 'BPV jump' (as defined in Barndorff-Nielsen and Shephard (2004)), 'negative RV' and 'positive RV' (as defined in Patton and Sheppard (2015)), and 'realised quarticity' (as defined in Bollerslev et al. (2016)) are described in Section 2 previously. The second column ('#') lists the number of variables included. The formula used to compile the defined variable is shown in the last column ('Characteristic'). This group has six defined variables in total. The most commonly used 5-minute sampling frequency is used for calculating these variables.

Table 2: HAR-family variables

Description	#	Characteristic
RV	1	$RV_t \equiv \sum_{i=1}^{M} r_{t,i}^2$
BPV	1	$BPV_t = \frac{1}{\mu_1^2} \sum_{i=1}^{M-1} \lvert r_{t,i} \rvert \lvert r_{t,i+1} \rvert$
BPV jump	1	$J_t = \max(RV_t - BPV_t, 0)$
Negative; positive RV	2	$RV_t^+ \equiv \sum_{i=1}^{M} r_{t,i}^{2+}$; $RV_t^- \equiv \sum_{i=1}^{M} r_{t,i}^{2-}$
Realised quarticity	1	$RQ_t \equiv (\frac{M}{3}) \sum_{i=1}^{M} r_{t,i}^4$

Notes: The first column ('Description') contains the name of the variables. Section 2 describes 'RV', 'BPV' and 'BPV jump' (as defined in Barndorff-Nielsen and Shephard (2004)), 'negative RV' and 'positive RV' (as defined in Patton and Sheppard (2015)), and 'realised quarticity' (as defined in Bollerslev et al. (2016)). The second column ('#') lists the number of variables included. The formula to compile the defined variable is shown in the last column ('Characteristic'). This group has six defined variables in total. The most commonly used 5-minute sampling frequency is used for calculating these variables.

3.2 News Variables

The *Dow Jones Newswires* database covers the Wall Street Journal, MarketWatch, Barron's news, etc. In this database, every news story is tagged with 'significant', 'about', or 'mention'. 'Significant' denotes a news story that is important to a specific ticker; 'about' denotes a news story about a ticker but of no particularly significance, while 'mentioned' denotes cases where the ticker is referenced but is not the main subject of the news story. As 'Significant' is introduced much later and is not available for the most part of our sample period, the tag 'about' is used for extracting company-related news for constructing the nine daily news variables listed in Table 3. Apart from 'News count', the other sentiment variables, 'negative', 'positive', 'uncertainty', 'litigious', 'weak modal', 'moderate modal', 'strong modal', and 'constraining', are compiled according to the LM dictionary (Loughran and McDonald, 2011).[1]

The steps for preprocessing the news data and calculating the sentiment variables follow those in Loughran and McDonald (2011). The sentiment measure for a particular company on day t is equal to the weighted word counts summed over all relevant words and all related news stories published about the company on the day t. The averaging is done across sentiment measures of all news stories for a company on a specific day. The proposed term-frequency-inverse-document-frequency (tf.idf) weighting scheme in Loughran and McDonald (2011) and Loughran and McDonald (2016) is applied when calculating the news sentiment variables to take into account that some words appear more often than others.[2]

[1]The LM dictionary is downloaded from Software Repository for Accounting and Finance, University of Notre Dame.

[2]A note of caution is due here since the LM dictionary was explicitly developed in the context of 10-K filings. Therefore its direct usage in other types of financial textual datasets may not reach the desired results (see Loughran and McDonald (2020) for more details.). Bearing in mind this limitation, the LM dictionary is

Table 3: News variables

Description	#	Characteristic[a]
News count	1	*Number of news*
Positive sentiment	1	*Average of positive sentiments*
Negative sentiment	1	*Average of negative sentiments*
Uncertainty sentiment	1	*Average of uncertainty sentiments*
Litigious sentiment	1	*Average of litigious sentiments*
Weak modal sentiment	1	*Average of weak modal sentiments*
Moderate modal sentiment	1	*Average of moderate modal sentiments*
Strong modal sentiment	1	*Average of strong modal sentiments*
Constraining sentiment	1	*Average of constraining sentiments*

Notes: The first column ('Description') contains the variable names. The second column ('#') lists the number of variables. The formula to compile the defined variable is shown in the third column ('Characteristic'). All sentiment measures in this table are calculated based on the LM dictionary (Loughran and McDonald, 2011). [a] 'News count' is the number of stories on that day. The average value is the daily average of the specific sentiment measure in the first column for all news stories for a company on that day.

3.3 Limit Order Book Variables

The *LOBSTER* dataset contains, for all NASDAQ stocks, LOB data and message data, the latter of which includes executions, submission, cancellation, and deletion of orders. The data cleaning steps are described in Appendix A, and the *CRSP* database is used to correct stock price and volume for stock splits, stock dividends, spin-offs, stock distributions, and right issues. Table 4 lists the LOB variables as defined in Kercheval and Zhang (2015) for up to $N = 10$ LOB levels giving rise to 132 variables in total, representing one of the most comprehensive lists covering both time-sensitive and time-insensitive LOB variables. The power and flexibility of ML make it possible to include such a large and comprehensive set of LOB variables in a single model.

4 Machine Learning

Recurrent neural networks (RNN), of which long short-term memory (LSTM) is the most important and well-known subset, are a type of artificial neural network designed to recognize patterns in data sequences that have a temporal dimension, such as financial times series. Subsection 4.1 below gives a brief review of RNN and its extension, LSTM, in Subsection 4.2. Subsection 4.3 discusses the regularisation techniques implemented here to prevent these ML

still the most reliable choice for sentiment analysis in finance at the time of writing.

Table 4: Limit order book variables

Description	#	**Characteristic**[a]	**Parameter**
Bid-ask spreads	10	$[(P_i^{ask} - P_i^{bid})]_{l=1}^{N}$	-
Mid prices	10	$[(P_i^{ask} + P_i^{bid})/2]_{l=1}^{N}$	-
Price differences	18	$[P_N^{ask} - P_1^{ask}, P_N^{bid} - P_1^{bid}]_{l=1}^{N}$	-
Absolute price differences	18	$[\lvert P_{l+1}^{ask} - P_i^{ask}\rvert, \lvert P_{l+1}^{bid} - P_i^{bid}\rvert]_{l=1}^{N}$	-
Mean prices	2	$[\frac{1}{n}\sum_{l=1}^{N} P_i^{ask}, \frac{1}{n}\sum_{l=1}^{N} P_i^{bid}]$	-
Mean volumes	2	$[\frac{1}{n}\sum_{l=1}^{N} V_i^{ask}, \frac{1}{n}\sum_{l=1}^{N} V_i^{bid}]$	-
Price/volume accumulated differences	2	$[\sum_{l=1}^{N}(P_i^{ask} - P_i^{bid}), \sum_{l=1}^{N}(V_i^{ask} - V_i^{bid})]$	-
Price/volume derivatives	40	$[dP_i^{ask}/dt,\ dP_i^{bid}/dt,\ dV_i^{ask}/dt,\ dV_i^{bid}/dt]$	$dt = 1day$
Average intensity[b]	10	$[\lambda_{\Delta t}^{(E)bid(ask)}, \lambda_{\Delta t}^{(S)bid(ask)}, \lambda_{\Delta t}^{(C)bid(ask)}, \lambda_{\Delta t}^{(D)bid(ask)}]$	$\Delta_t = 1day$
Relative intensity[c]	10	$[1_{\{\lambda_{\Delta t}^{(E)bid(ask)} > \lambda_{\Delta T}^{(E)bid(ask)}\}}, 1_{\{\lambda_{\Delta t}^{(S)bid(ask)} > \lambda_{\Delta T}^{(S)bid(ask)}\}}, 1_{\{\lambda_{\Delta t}^{(C)bid(ask)} > \lambda_{\Delta T}^{(C)bid(ask)}\}}, 1_{\{\lambda_{\Delta t}^{(D)bid(ask)} > \lambda_{\Delta T}^{(D)bid(ask)}\}}]$	$\Delta_T = 1day$, $\Delta_t = 15mins$
Accelerations[d]	10	$[d\lambda^{(E)bid(ask)}/dt,\ d\lambda^{(S)bid(ask)}/dt,\ d\lambda^{(C)bid(ask)}/dt,\ d\lambda^{(D)bid(ask)}/dt]$	$dt = 1day$

Notes: This table contains the LOB variables in Kercheval and Zhang (2015). The first column ('Description') contains the variable names. The second column ('#') lists the number of variables included from 10 LOB levels. The formula to compile the defined variable is shown in the third column ('Characteristic'). The fourth column ('Parameter') specifies the time parameter. When there is no time parameter, the last snapshot of the LOB for a particular day is used. [a] (E), (S), (C), and (D) stand for execution, submission, cancellation, and deletion of orders, respectively. 'P' and 'V' stand for price and volume.
[b] The ratio of the defined variable in the nominator to the total number of orders for that day.
[c] This value is a one-or-zero binary number. [d] The nominators are from 'Average Intensity'.

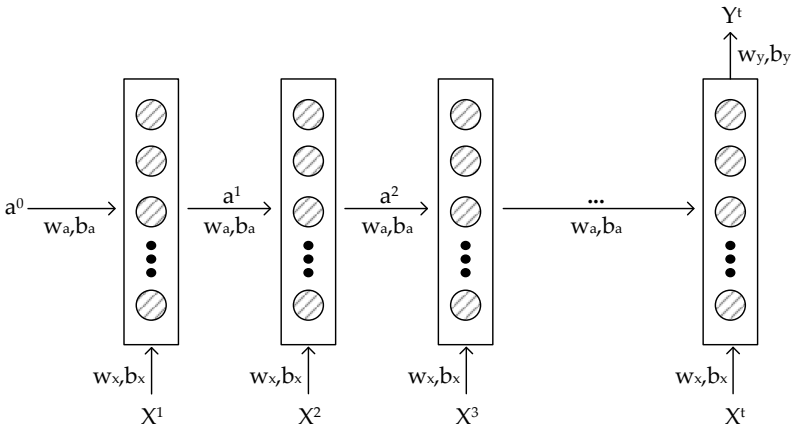


Figure 1: Recurrent neural network abstract representation

Notes: This representation of the RNN has t input vectors through time (X^1,X^2,X^3,...,X^t) and one output (Y^t). W_x and b_x are the shared weights and biases between inputs and the neural network layers for time step 1 to t, W_a and b_a are the shared weights and biases between different layers, and W_y and b_y are the shared weights and biases between the last layer of neural network and the single output (Y^t). For the east of exposition, this illustration portrays an RNN for a single input variable. Also, in this representation, every layer has an arbitrary number of units which are demonstrated by the hatched circles, a^1 to a^t are the transferred information from one layer to the subsequent layer, and a^0 is the initial input vector for the first layer.

models from over-fitting, and Subsection 4.4 presents the structure of the proposed ML models used in this study.

4.1 Recurrent Neural Network

Figure 1 is a representation of the RNN with t input vectors (X^1,X^2,X^3,...,X^t) and one output (Y^t). W_x and b_x are the shared weights and biases between inputs and the neural network layers for time step 1 to t, W_a and b_a are the shared weights and biases between layers, and W_y and b_y are the shared weights and biases between the last layer of neural network and the single output (Y^t). For the east of exposition, Figure 1 portrays an RNN for a single input variable. Also, in this representation, every layer has an arbitrary number of units[1] which are demonstrated by the hatched circles, a^1 to a^t are the transferred information from one layer to the subsequent layer, and a^0 is the zero input vector for the first layer. a^t is defined as follows:

$$a^t = tanh(w_{ax}[a^{t-1}, X^t] + b_{ax}), \tag{8}$$

where w_{ax} is the stacked matrix of w_a and w_x, b_{ax} is the stacked matrix of b_a and b_x, and $tanh$ is the hyperbolic tangent activation function. Also, the output (Y^t) is defined as follows:

[1] A unit (neuron) is a basic building block that takes in a set of inputs, performs mathematical operations on them, and produces a single output. Generally, an ML model with more units can be considered more complex.

$$Y^t = tanh(w_y a^t + b_y), \tag{9}$$

where w_y and b_y are the output weights and biases, and $tanh$ is the hyperbolic tangent activation function.

The weights and biases in Equation (8) and Equation (9) are estimated based on minimising MSE using gradient descent with backpropagation. The RNN suffers from the problem of vanishing gradient because of the way in which the sequential data is modelled through time. The impact of, e.g., X_1 and a_1 on a_T and Y_t become very small or near zero for $w < 1$, and large t. This is the fundamental weakness of RNN; it is not very good at capturing long-dependency in data with a long sequence (Hochreiter, 1998). Subsection 4.2 below introduces LSTM, an extended RNN for mitigating this weakness.

4.2 Long Short-Term Memory

Hochreiter and Schmidhuber (1997) use gated cells with their own sets of weights to store, read or erase historical information. To tackle the vanishing gradient problem, a more constant error rate is maintained in LSTM to allow the model to continue learning over many steps. The LSTM unit consists of a cell, an input gate, an output gate, and a forget gate. The cell remembers values over arbitrary time intervals, and the three gates regulate the flow of information into and out of the cell. The goal is to provide a memory that can last after many time steps. First define the candidate memory cell ($\tilde{c}^t$) as:

$$\tilde{c}^t = tanh(w_c[a^{t-1}, X^t] + b_c), \tag{10}$$

where w_c and b_c are the weights and biases of the candidate memory cell, and $tanh$ is the hyperbolic tangent activation function. Then, the update, forget, and output gates are defined as:

$$G_u = \sigma(w_u[a^{t-1}, X^t] + b_u), \tag{11}$$

$$G_f = \sigma(w_f[a^{t-1}, X^t] + b_f), \tag{12}$$

$$G_o = \sigma(w_o[a^{t-1}, X^t] + b_o), \tag{13}$$

where w_u and b_u are the weight and bias of the update gate, w_f and b_f are the weight and bias of the forget gate, w_o and b_o are the weight and bias of the output gate, and σ is the sigmoid

activation function. Taken together, the updated memory cell (c^t) is calculated as:

$$c^t = G_u \times \tilde{c}^t + G_f \times c^{t-1}, \tag{14}$$

where c^{t-1} is the memory cell at time $t-1$ and $\tilde{c}^t$ is the candidate memory cell at time t from Equation (10). The update gate controls the flow of the input activation into the memory cell, while the forget gate scales the internal state of the cell before adding it to the memory cell. Finally, a^t, the control for information flowing into the next layer, is calculated as:

$$a^t = G_o \times tanh(c^t), \tag{15}$$

where G_o is the output gate in Equation (13) and c^t is the memory cell in Equation (14) and $tanh$ is the hyperbolic tangent activation function. The role of the output gate is to control the output flow of the cell activation into the rest of the neural network. All the weights and biases are to be learned during training. Collectively, LSTM avoided the vanishing gradient by suitably memorizing and forgetting some past states, so theoretically, it can capture long-dependencies in the sequential data.

4.3 Regularisation

One typical issue with ML is over-fitting, i.e., when the error rate in the training set is artificially low, but the error rate in the test data is high. Over-fitting is not confined to ML models only. However, because of the typically large input dataset and complex structure, ML models are prone to over-fitting, which we plan to curtail using regularisation techniques. In this study, we implemented L^2 regularisation and dropout, two of the most widely used regularisation techniques in recent years.

The L^2 regularisation works by adding a regularisation term, $\lambda||w||^2$, to the loss function, $\mathcal{L}$, as follows:

$$\mathcal{L}'(w; X, Y) = \mathcal{L}(w; X, Y) + \lambda||w||^2, \tag{16}$$

where, $\mathcal{L}$ and $\mathcal{L}'$ are the initial and the modified loss functions, w, X, and Y are, respectively, the weights, inputs, and output of ML model, $||.||^2$ is the L^2 norm, and λ is the regularisation factor. The regularisation term penalises a model with larger weights. The larger the regularisation factor, λ, the more severe the penalty. Together, $\lambda||w||^2$ prevents the ML model from over-fitting. Loshchilov and Hutter (2017) claimed this regularisation technique is beneficial for the stochastic gradient descent optimisation algorithms, such as the adaptive gradient algorithms used in this study.

Dropout works by removing some randomly selected units and their incoming and outgoing connections during the training process. As a rule of thumb, the optimal dropout rate typically ranges between 20% and 50% of the input and hidden units (Srivastava et al., 2014). The purpose of dropout is to add noise to the neural network optimisation, making the training process more difficult and decreasing the amount of over-fitting.

4.4 ML Model Structure

In this study, we follow the classical framework to evaluate forecasting performance by producing out-of-sample forecasts using only *ex-ante* information. This means repeatedly re-estimating the ML model using only historical information available at a time and rolling forward one step at a time over the out-of-sample period. Such a process is extremely laborious, even without the complexity of testing different ML models using many combinations of hyperparameters and 147 input variables for each of the 23 stocks. Instead of testing several ML models, we have decided here to focus on only LSTM and concentrate on gaining a deeper understanding of its strength and weakness in volatility forecasting. Recent studies have found NN models to be superior among some popular ML models, and among NN models, LSTM has structure resemblance a time series model. We believe that a detailed investigation of LSTM is crucial at this juncture for its importance in time series analysis. Furthermore, an even more essential objective revolves around assessing the importance of the model's ability to forecast RV using a rich and diverse set of predictors in a single setting.

As explained in Section 1 and Subsection 4.2, the LSTM has many unique features making it ideal for handling sequential data, including financial time series. This model is one of the top ML choices for modelling time series data by academia and industry. Hence, our goal here is to focus on the LSTM, testing a large variation of hyperparameters, in order to understand better the tuning strategy for achieving the best possible performance. We also believe testing many models with limited variations in tuning parameters could lead to misleading conclusions. Furthermore, we have chosen a single-layer LSTM instead of a more complex LSTM. A simple vanilla LSTM is chosen as it is more tractable for follow-up analyses. A simpler LSTM model is easy to train with a smaller dataset; substantially more time, computing capacity and data are needed for training a more complex LSTM model. Finally, complex ML models have many more combinations of hyperparameters to test and choose from, making the analyses more complicated and less intuitive.[1]

Figure 2 presents the structure of the LSTM model used in this study. For every explanatory

[1]Although it is not clear the theoretical benefit of a deeper sequential architecture, deeper models could lead to better performance in some specific tasks (Goldberg, 2016). With our objectives in mind and all the constraints at hand, a multi-layer (stacked) model is left for future research.

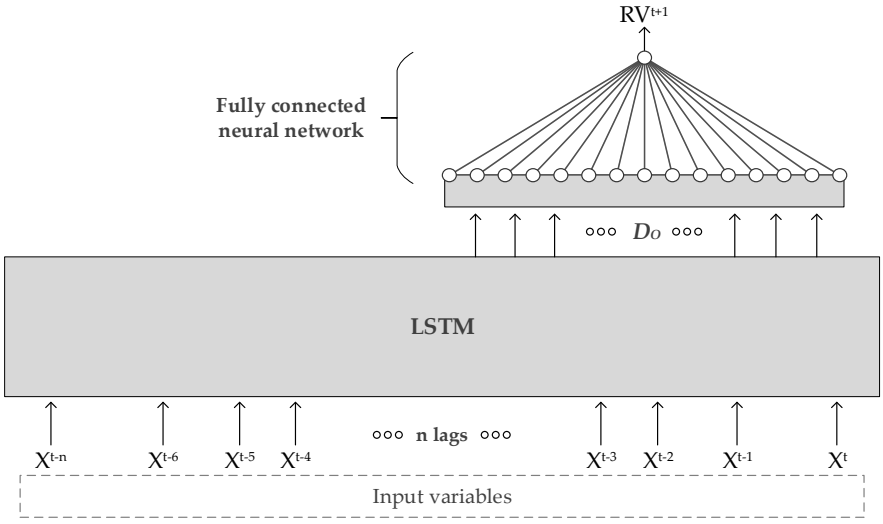


Figure 2: ML model structure

Notes: For every input variable (X), n lags of that variable (X^t,X^{t-1},...,X^{t-n}) are included as input variables to the LSTM model. The number of outputs of the LSTM (D_O) is equal to the number of units in the LSTM model. A fully connected neural network (FCNN) converts D_O outputs of the LSTM to a single RV^{t+1}.

or predictive variable (X), n lags of that variable (X^t,X^{t-1},...,X^{t-n}) are included as input variables to the LSTM model. The number of outputs from the LSTM (D_O) is equal to the number of units in this model. A fully connected neural network (FCNN) is used for converting D_O outputs into a single forecast (RV^{t+1}). Two regularisation techniques, L^2 regularisation and dropout as described in Subsection 4.3, are implemented to avoid over-fitting.

The ML technical specifications implemented here are as follows: the optimisation algorithm is ADAM (Kingma and Ba, 2014). ADAM is an adaptive learning rate optimisation algorithm based on stochastic gradient descent. The learning rate and learning rate decay[1] are set to 0.001 and 10^{-5}, respectively. For both the LSTM and FCNN, the kernel and bias L^2 regularisation techniques are applied with 10^{-4} as the decay parameter. The second regularisation technique, dropout, is applied between the LSTM and FCNN with the rate of 0.5. For LSTM, the sigmoid activation function is chosen for the cell and hidden states, and the tangent hyperbolic activation function is chosen for the input, forget, and output gates. Also, for FCNN, the rectifier activation function[2] is chosen.[3] For the primary experiments in Section 5, minimising MSE is chosen as the loss function, and the number of epochs is set to 50. The LSTM is tested with

[1]The decay parameter of the Adam optimizer governs the learning rate scheduling and it plays a role in refining the optimization procedure.

[2]A unit with the rectifier activation function is called a rectified linear unit (ReLU).

[3]The choice of hyperparameters adopted here is consistent with the default values used in the literature and common practices. Certain hyperparameters, like the learning rate decay and decay parameter, are chosen through trial-and-error. While the robustness checks in Section 7 provide valuable insights into the sensitivity of the choice of loss function, and the number of units and epochs, there are other hyperparameters that could significantly impact on model performance. Because of the extensive number of models trained in this study and the computational intensity involved, investigating the simultaneous impact exerted by all hyper-parameters is beyond the scope of this research. Therefore, a potential avenue for future research would be to extend the robustness analysis to include a more comprehensive examination of the model performance sensitivity to hyperparameters' choice.

Table 5: Number of independent variables and parameters of models

Group	**HAR-family of models**							
Model	AR	HAR	HAR-J	CHAR	SHAR	ARQ	HARQ	HARQ-F
Number of variables	1	3	4	3	4	2	4	6
Number of parameters	1	4	5	4	5	3	5	7

Group	**ML models**[a]			
Model	HAR-ML	News-ML	OB-ML	News/OB-ML
Number of variables	6 × NoL[b]	15 × NoL	138 × NoL	147 × NoL
Number of parameters (5 units)[c]	246	426	2886	3066
Number of parameters (10 units)	691	1051	5971	6331
Number of parameters (15 units)	1336	1876	9256	9796
Number of parameters (20 units)	2181	2901	9256	13461
Number of parameters (25 units)	3226	4126	16426	17326

Notes: HAR-ML contains all HAR-family variables (described in Subsection 3.1). News-ML contains all news variables (described in Subsection 3.2). OB-ML contains all LOB variables (described in Subsection 3.3), and finally, News/OB-ML includes all three sets of variables. [a] All ML models contain the HAR-family variables. [b] Number of lags (i.e., 21 for the primary experiments). [c] Number of units in the LSTM model.

5, 10, 15, 20, and 25 units. Here, the FCNN has one layer (1 unit).

For the primary experiments in Section 5, the use of 21 lags (i.e., $n = 20$ in Figure 2) for all input variables is consistent with the HAR model structure. However, the number of variables included in the ML model is far greater than that included in HAR, even with the restriction to 21 lags. In the second stage (i.e., robustness checks described in Section 7), we examine the sensitivity of our findings against the number of input variables. Finally, all predictive variables are standardised by removing the mean and scaling to unit variance. The standardisation is executed every day when the window is rolled forward using only the data in the relevant window[1].

Table 5 lists the number of variables and the number of parameters for all HAR-family of models in the top panel and all variants of ML models tested here in the bottom panel. It is clear that the ML models use far more input variables than the HAR-family of models. Combining all datasets, the News/OB-ML model has 147 × 21 input variables compared to only one input variable in the AR model. The ability to deal with nonlinear relationships in a high-dimensional environment is a vital feature of ML models. Note also that adding more lags to the ML models does not change the number of parameters in the LSTM model; only the number of variables and the number of units determine the number of parameters. For each of the 23 tickers, only the data for that ticker is used for training. The same seed in the random number generator (RNG) is used for all models to ensure reproducible results.

[1]In accordance with the HAR model structure, the window size for the primary experiment is set to 21. For the robustness checks in Subsection 7.1, we experimented with window size set to 5 and 1.

5 Out-of-Sample Forecasting Results

The entire sample period of 27 July 2007 to 27 January 2022 was separated into an in-sample training period from 27 July 2007 to 11 September 2015 (2046 days) and an out-of-sample forecasting period from 14 September 2015 to 27 January 2022 (1604 days).[1] Following Poon and Granger (2003), the model's forecasting power is judged only according to the out-of-sample forecasting performance. To better understand the forecasting performance on normal vs high volatility days, we identify high volatility days using simple nonparametric criteria, i.e., whenever the RV is greater than $Q3 + 1.5 \times IQR$, where $Q1$ and $Q3$ are the first and third quantiles, and IRQ is equal to $Q3 - Q1$. All these values are separately calculated for every stock using only the data in the out-of-sample period. Based on this definition and averaging across all stocks, about 10% (160 days) of the out-of-sample period (1604 days) are classified as high volatility days. We have purposely avoided parametric jump estimation as it is sensitive to the assumption of the stock price dynamics and the bandwidth adopted data frequency.[2] We consistently noticed that the forecast of normal and high volatility requires different ML specifications and different information sets. So as long as the out-of-sample forecasts are separated into a main set of normal observations and a smaller set of extreme observations (i.e., high volatility days), disregarding the method used to separate them, our findings and conclusion should continue to hold.

We use reality check (RC) to test the forecasting performance of the ML model against every model in the HAR-family. In line with White (2000) and Bollerslev et al. (2016), the stationary bootstrap of Politis and Romano (1994) with 999 re-samplings and the average block length of 5 are used for this test.[3] The hypotheses of the test are defined as follows:

$$\begin{aligned} H_0 &: \underset{k=1,\ldots,n}{Min} \ \mathbb{E}[L^k(RV, X) - L^0(RV, X)] \leq 0, \\ H_1 &: \underset{k=1,\ldots,n}{Min} \ \mathbb{E}[L^k(RV, X) - L^0(RV, X)] > 0, \end{aligned} \tag{17}$$

where L^k is the loss measure of the k^{th} out of n benchmark HAR models, and L^0 is the loss measure of the desired ML model. A rejection of the null hypothesis means $L^k \leq L^0$, and the ML model outperformed the k^{th} HAR model.

Following Patton (2011), the MSE and QLIKE loss functions are chosen for measuring the RV

[1] We need to have similar data characteristics embedded in both in-sample estimation and out-of-sample forecasting. Hence, the selection of in-sample and out-of-sample periods here is also driven by the desire to includes days with extreme volatility in both periods. In particular, the in-sample period encompasses the period of 2008 financial crises, while the out-of-sample period includes the 2019 COVID-19 disruptions.

[2] The distinction between low and high RVs is not known a-priori; it is solely made ex-post for the purpose of comparing out-of-sample performance.

[3] Our analysis shows that the results are not sensitive to the choice of block length.

forecasting performance in addition to MDA (Mean Directional Accuracy). According to Patton (2011), the MSE and QLIKE loss functions are among the family of robust and homogeneous loss functions for volatility forecasting comparison.[1] Rankings of volatility forecasts based on these loss functions are robust to noise in the proxy, and the rankings are invariant to the choice of units of measurement. On the other hand, MDA is a useful measure complementing these two cardinal measures, especially for monitoring the over-fitting of the models. The three loss functions are defined as follows:

$$MSE(RV_t, \widehat{RV_t}) \equiv \frac{1}{N}\sum_{t=1}^{N}(RV_t - \widehat{RV_t})^2, \tag{18}$$

$$QLIKE(RV_t, \widehat{RV_t}) \equiv \frac{1}{N}\sum_{t=1}^{N}\left(\frac{RV_t}{\widehat{RV_t}} - \log(\frac{RV_t}{\widehat{RV_t}}) - 1\right), \tag{19}$$

$$MDA(RV_t, RV_{t-1}, \widehat{RV_t}) \equiv \frac{1}{N}\sum_{t=1}^{N} 1_{sign(RV_t - RV_{t-1}) == sign(\widehat{RV_t} - RV_{t-1})}, \tag{20}$$

where RV_t is the true RV at time t, $\widehat{RV_t}$ is the forecast RV at time t, N is the number of days in the out-of-sample period, and $sign(\cdot)$ and 1 are the sign and indicator functions. Following Bollerslev et al. (2016), if the forecast RV is larger (smaller) than the maximum (minimum) of the RV in the estimation series, it is replaced by the average of the RV in the estimation series.

We compare the out-of-sample forecasting performance of four ML models, differ by their information sets against the benchmark CHAR model selected by Rahimikia and Poon (2022) as the best performing HAR model. If we define, for stock i, and for each ML model j,

$$\Delta_{MSE,i,j} = MSE_i^{ML}\ (Model_j) - MSE_i^{OLS}\ (CHAR), \ \ and \tag{21}$$

$$\Delta_{QLIKE,i,j} = QLIKE_i^{ML}\ (Model_j) - QLIKE_i^{OLS}\ (CHAR), and \tag{22}$$

$$\Delta_{MDA,i,j} = MDA_i^{ML}\ (Model_j) - MDA_i^{OLS}\ (CHAR), \tag{23}$$

where $j = 1,\ 2,\ 3,\ 4$ is one of the four ML models in Table 5, and $i = 1,\ \ldots,\ 23$ identifying each of the 23 stocks. For the ML models, $j = 1$ is HAR-ML, and the HAR-family variables are described in Subsection 3.1; $j = 2$ is News-ML, and the news variables are described in Subsection 3.2; $j = 3$ is OB-ML, and the LOB variables are described in Subsection 3.3, and finally $j = 4$ is News/OB-ML, i.e. the news and LOB variables combined. It is worth noting that News-ML, OB-ML and News/OB-ML also contain the HAR variables. A lower MSE and QLIKE indicate a better performance, and a lower MDA indicates a poorer performance. Hence, a negative $\Delta_{MSE/QLIKE}$ in Equation (21) and Equation (22) and a positive Δ_{MDA}

[1] See Poon and Granger (2003) for a review of the volatility performance metrics.

in Equation (23) indicate improvements from using ML; a positive (negative) MSE/QLIKE (MDA) value indicates ML model performance degradation.

Next, ML model $j's$ average $\Delta_{MSE/QLIKE/MDA,j}$ for the 23 stocks are:

$$Average\ \Delta_{MSE,j} = \frac{1}{23}\sum_{i=1}^{23} \Delta_{MSE,i,j}, \tag{24}$$

$$Average\ \Delta_{QLIKE,j} = \frac{1}{23}\sum_{i=1}^{23} \Delta_{QLIKE,i,j}, \tag{25}$$

$$Average\ \Delta_{MDA,j} = \frac{1}{23}\sum_{i=1}^{23} \Delta_{MDA,i,j}, \tag{26}$$

while $Median\ \Delta_{MSE,j}$, $Median\ \Delta_{QLIKE,j}$, and $Median\Delta_{MDA,j}$ are the median equivalent of Equation (24), Equation (25) and Equation (26), respectively.

5.1 Full Out-of-Sample Period

Table 6 reports the average and median $\Delta_{MSE/QLIKE/MDA}$ for the four ML models with 21 lags of every group of variables included and the different number of units (5, 10, 15, 20, and 25). The corresponding RC value is the percentage of tickers with better ML performance, in terms of MSE, QLIKE or MDA at the 5% and 10% significance levels against every model in the HAR-family of models (AR1, HAR, HAR-J, CHAR, SHAR, ARQ, HARQ, and HARQ-F). For cardinal forecasts evaluated using MSE and QLIKE, and considering the RC results, improvement in forecasting performance is clear, especially for HAR-ML and OB-ML groups. However, the results in Table 6 are mixed. In all cases and for all groups of ML models, the differences (both average and median values) show degradation in performance compared to the CHAR model as a benchmark. For directional forecasts evaluated using MDA, ML outperformed CHAR in most specifications, reaching the most remarkable improvement (around 4%) and 100% for RC in the News/OB-ML group. Taken together, except for the MDA loss function, consistent improvements are noticeable only from RC results.[1] However, by separating the out-of-sample actual RVs into normal and high volatility days, a clearer picture emerged in

[1]Table 6 indicates that, in the full out-of-sample period, ML underperformed in terms of average and median MSE and QLIKE but outperformed in terms of RC. Note that average and median MSE and QLIKE are calculated for ML against only CHAR, while RC is calculated for ML against all models in the HAR family, including CHAR. To further explore this finding, Figure B1 in Appendix B depicts the performance of OB-ML with 25 units (as the optimal choice according to Table 6) against AR1, HAR, HAR-J, CHAR, SHAR, ARQ, HARQ, and HARQ-F for the entire out-of-sample period. The left and right radar charts contain results for the MSE and QLIKE loss functions, respectively. The negative value shows improvement, and the bold circle inside these radar charts shows no improvement. These plots clearly demonstrate the superior forecasting performance of CHAR across most tickers for both MSE and QLIKE loss functions. This means that the performance of other models in the HAR-family of models was notably worse, contributing to the overall 'successful' RC results for ML. This finding is true not only for the OB-ML model with 25 units but also for all other ML model configurations outlined in Table 6.

Table 6: Out-of-sample volatility forecasting performance: 14 September 2015 to 27 January 2022 (1604 days)

			HAR-ML					News-ML					OB-ML					News/OB-ML				
	Units		5	10	15	20	25	5	10	15	20	25	5	10	15	20	25	5	10	15	20	25
Full out-of-sample period																						
MSE[a]		Avg	37.383	33.209	29.804	26.665	23.628	38.208	34.009	29.890	27.138	24.066	34.749	30.331	26.107	24.111	**22.420**	36.423	30.378	26.432	25.432	22.745
		Med	27.446	23.889	23.112	19.608	19.238	27.166	24.793	24.319	19.554	19.138	25.115	20.810	19.809	18.542	17.919	25.428	21.336	19.444	19.039	**17.591**
RC[d]	5%		13.04	34.78	39.13	56.52	73.91	17.39	26.09	39.13	52.17	73.91	26.09	39.13	56.52	73.91	**78.26**	17.39	39.13	47.83	69.57	**78.26**
	10%		47.83	60.87	82.61	86.96	**100**	52.17	60.87	82.61	91.30	95.65	60.87	73.91	91.30	**100**	**100**	65.22	69.57	86.96	91.30	95.65
QLIKE[b]		Avg	3.757	1.544	0.823	0.402	**0.214**	3.669	1.759	0.889	0.544	0.281	2.016	0.710	0.354	0.279	**0.214**	2.761	0.648	0.360	0.335	0.245
		Med	3.317	1.317	0.646	0.353	**0.158**	3.563	1.791	0.750	0.456	0.190	1.064	0.486	0.288	0.234	0.174	1.812	0.413	0.280	0.264	0.202
RC	5%		0.00	0.00	0.00	0.00	**13.04**	0.00	0.00	0.00	4.35	8.70	0.00	4.35	0.00	8.70	**13.04**	0.00	0.00	0.00	0.00	8.70
	10%		0.00	4.35	17.39	34.78	**60.87**	0.00	0.00	4.35	8.70	43.48	4.35	8.70	13.04	17.39	21.74	0.00	8.70	13.04	21.74	17.39
MDA[c]		Avg	-0.008	0.005	0.017	0.020	0.019	-0.008	0.001	0.016	0.022	0.025	0.021	0.035	**0.040**	0.037	0.028	0.006	0.037	**0.040**	0.036	0.031
		Med	-0.001	0.007	0.021	0.021	0.019	0.001	0.006	0.019	0.022	0.025	0.032	0.035	0.037	0.034	0.027	0.018	0.041	**0.046**	0.036	0.030
RC	5%		65.22	91.30	95.65	**100**	**100**	65.22	82.61	95.65	95.65	**100**	78.26	91.30	**100**	**100**	**100**	69.57	91.30	**100**	**100**	**100**
	10%		69.57	95.65	95.65	**100**	**100**	65.22	86.96	95.65	**100**	**100**	86.96	95.65	**100**	**100**	**100**	73.91	91.30	**100**	**100**	**100**

Notes: The best value in each row is marked in bold.
[a] The difference between the two mean (or median) MSEs for ML and CHAR for 23 tickers (a negative value indicates improvement, and a positive value indicates degradation in performance).
[b] The difference between the two mean (or median) QLIKEs for ML and CHAR for 23 tickers (a negative value indicates improvement, and a positive value indicates degradation in performance).
[c] The difference between the two mean (or median) MDAs for ML and CHAR for 23 tickers (a positive value indicates improvement, and a negative value indicates degradation in performance).
[d] Percentage of tickers with the outstanding performance of ML against the HAR-family of models at the 5% and 10% significance levels.

Subsection 5.2 below.

5.2 Normal vs High Volatility Days

Table 7 reports the results for normal and high volatility days separately. The first paragraph of Section 5 explains how out-of-sample period actual RVs are separated into normal vs high volatility days. First, consider the top panel of Table 7 for normal volatility days, which on average constitute 90% of the out-of-sample forecast period. The four ML models outperformed the CHAR model as average/median MSE/QLIKE values are mostly negative. When the MSE measure is used, except in one case, the RC values are all 100%. When QLIKE is used, in many cases, RC is near 100%. This is strong evidence supporting ML's forecasting power against all HAR-family of models for normal volatility days. The substantial improvement is also evident for the MDA loss function, increasing the directional forecasting performance by around 7% in the best-performing model. Across the four ML models, OB-ML is the best-performing group of models. In general, the higher number of units, the better the forecasting performance for normal volatility days.

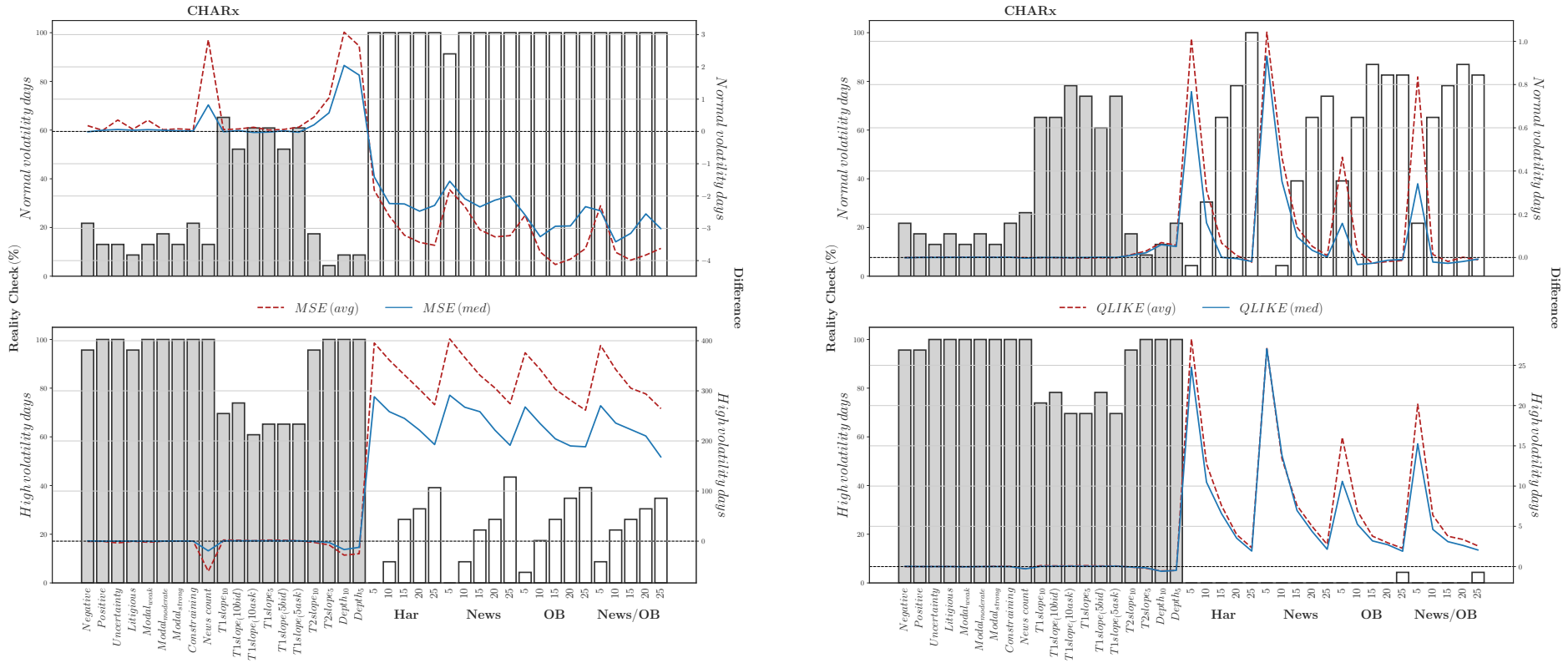
The results for high volatility days in the bottom panel of Table 7 present a completely different picture. All four ML models underperformed the CHAR model as average/median MSE/QLIKE values are all positive. When the MSE measure is used, the RC values are low, most of which are below 50%, and even worse when QLIKE is used. Likewise, the MDA loss function results show deterioration in the forecasting performance. These sharply contrasting results to normal volatility days suggest that the ML models fitted to the full in-sample period dominated by normal volatility days are not appropriate for forecasting the infrequent high volatility days. One possible explanation for these contrasting results could be due to RV's persistence differing on high vs normal volatility days. It is widely documented that volatility is highly persistent, but not when volatility is extreme. Volatility half-life is very much shorter for extreme volatility compared to the normal level of volatility.[1] This is the first and most important clue that separate modelling considerations are needed for normal vs high volatility regimes. Here, as we did not separate the normal vs high volatility days, the optimisation was driven by the 90% of normal volatility days forecast evaluations. Nevertheless, although the high volatility days constitute only 10% of the out-of-sample period, the poor performance of the ML models during high volatility days strongly indicates that these models, particularly in their basic form without additional adjustments to RV definition and/or the model itself, might be economically less useful. This is because precise forecasts of high volatility days hold greater importance in a risk management scenario compared to days with low volatility. Our

[1] In the individual stocks analyses and the robustness checks later, we also find very different ML modelling considerations for normal vs high volatility days.

Table 7: Out-of-sample volatility forecasting performance: Normal vs high volatility days
14 September 2015 to 27 January 2022 (1604 days)

			HAR-ML					News-ML					OB-ML					News/OB-ML				
		Units	5	10	15	20	25	5	10	15	20	25	5	10	15	20	25	5	10	15	20	25
Normal volatility days																						
MSE[a]		Avg	-1.823	-2.624	-3.213	-3.434	-3.526	-1.806	-2.323	-3.045	-3.264	-3.227	**-2.598**	**-3.733**	**-4.122**	**-3.954**	**-3.627**	-2.301	-3.742	-3.985	-3.827	-3.622
		Med	-1.408	-2.232	-2.245	-2.469	-2.290	-1.542	-2.085	-2.336	-2.128	-1.998	**-2.590**	**-3.259**	**-2.937**	**-2.926**	**-2.330**	-2.459	-3.419	-3.155	-2.553	-3.014
RC[d]	0.05		100	100	100	100	100	91.30	100	100	100	100	**100**	**100**	**100**	**100**	**100**	100	100	100	100	100
	0.10		100	100	100	100	100	91.30	100	100	100	100	**100**	**100**	**100**	**100**	**100**	100	100	100	100	100
QLIKE[b]		Avg	1.012	0.312	0.067	0.009	-0.022	1.042	0.463	0.140	0.052	0.009	**0.464**	**0.033**	**-0.027**	**-0.019**	**-0.014**	0.835	0.014	-0.019	0.001	-0.011
		Med	0.767	0.160	0.000	-0.005	-0.018	0.931	0.353	0.096	0.034	0.002	**0.158**	**-0.033**	**-0.027**	**-0.013**	**-0.009**	0.341	-0.021	-0.027	-0.018	-0.007
RC	0.05		4.35	30.44	65.22	78.26	100	0.00	4.35	39.13	65.22	73.91	**39.13**	**65.22**	**86.96**	**82.61**	**82.61**	21.74	65.22	78.26	86.96	82.61
	0.10		4.35	34.78	69.57	82.61	100	0.00	4.35	39.13	78.26	86.96	**39.13**	**65.22**	**86.96**	**82.61**	**82.61**	21.74	73.91	82.61	91.30	86.96
MDA[c]		Avg	0.028	0.036	0.046	0.044	0.039	0.028	0.034	0.046	0.049	0.047	**0.061**	**0.072**	**0.070**	**0.063**	**0.049**	0.045	0.074	0.070	0.064	0.052
		Med	0.029	0.034	0.046	0.042	0.038	0.037	0.040	0.045	0.050	0.049	**0.065**	**0.067**	**0.070**	**0.065**	**0.050**	0.063	0.079	0.071	0.061	0.052
RC	0.05		86.96	95.65	100	100	100	82.61	91.30	100	100	100	**100**	**100**	**100**	**100**	**100**	91.30	100	100	100	100
	0.10		86.96	95.65	100	100	100	86.96	91.30	100	100	100	**100**	**100**	**100**	**100**	**100**	95.65	100	100	100	100
High volatility days																						
MSE		Avg	395.377	361.209	331.745	302.807	272.308	403.591	366.455	331.117	305.635	274.205	375.968	342.720	303.138	281.442	261.406	390.463	342.752	305.256	293.806	264.466
		Med	288.174	258.327	244.925	221.391	192.750	291.257	267.271	258.363	221.423	191.293	267.776	234.378	204.224	190.029	188.360	270.294	235.611	222.772	209.848	167.810
RC	0.05		0.00	8.70	26.09	30.44	39.13	0.00	8.70	21.74	26.09	43.48	4.35	17.39	26.09	34.78	39.13	8.70	21.74	26.09	30.44	34.78
	0.10		30.44	39.13	47.83	56.52	69.57	26.09	39.13	39.13	47.83	73.91	30.44	43.48	52.17	56.52	69.57	26.09	43.48	52.17	60.87	65.22
QLIKE		Avg	28.280	12.758	7.556	3.968	2.335	27.096	13.419	7.579	4.993	2.750	16.055	6.912	3.781	2.964	2.268	20.173	6.376	3.768	3.366	2.560
		Med	24.764	10.486	6.566	3.526	1.925	27.049	13.834	6.969	4.367	2.148	10.586	5.262	3.191	2.721	1.906	15.242	4.617	3.098	2.627	2.054
RC	0.05		0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	4.35	0.00	0.00	0.00	0.00	4.35
	0.10		0.00	4.35	4.35	0.00	13.04	0.00	0.00	0.00	4.35	17.39	4.35	4.35	0.00	4.35	4.35	4.35	0.00	0.00	4.35	4.35
MDA[b]		Avg	-0.035	-0.033	-0.032	-0.030	-0.032	-0.035	-0.033	-0.032	-0.030	-0.029	-0.035	-0.033	-0.032	-0.030	-0.027	-0.035	-0.033	-0.030	-0.028	-0.025
		Med	-0.029	-0.029	-0.029	-0.024	-0.026	-0.029	-0.029	-0.029	-0.026	-0.026	-0.029	-0.029	-0.027	-0.027	-0.026	-0.029	-0.029	-0.027	-0.029	-0.026
RC	0.05		0.00	0.00	0.00	4.35	8.70	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	4.35
	0.10		0.00	0.00	0.00	4.35	13.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	4.35

Notes: The best performing group of variables are marked in bold.
[a] The difference between the two mean (or median) MSEs for ML and CHAR for 23 tickers (a negative value indicates improvement, and a positive value indicates degradation in performance).
[b] The difference between the two mean (or median) QLIKEs for ML and CHAR for 23 tickers (a negative value indicates improvement, and a positive value indicates degradation in performance).
[c] The difference between the two mean (or median) MDAs for ML and CHAR for 23 tickers (a positive value indicates improvement, and a negative value indicates degradation in performance).
[d] Percentage of tickers with the outstanding performance of ML against the HAR-family of models at the 5% and 10% significance levels.



(a) Forecast evaluation under MSE

(b) Forecast evaluation under QLIKE

Notes: The bar chart is the percentage of tickers with the outstanding performance considering the MSE loss function in Figure 3a and the QLIKE loss function in Figure 3b at the 5% significance level of the RC compared to the all HAR-family of models as the benchmark for every specified extended CHAR model (hatched bars) and the ML model (white bars). The values for the bar chart can be read from the left-hand axis. The dashed (solid) line represents the difference between the average (median) of the out-of-sample MSEs and QLIKEs of the specified model with the CHAR model for 23 tickers (negative value shows improvement, and positive value shows degradation in performance). The values for the dashed and solid lines can be read from the right-hand axis. The horizontal dashed line represents no improvement.

Figure 3: ML models and CHARx models comparison

findings show that ML models trained using the full in-sample period did not perform well on 10% of (non-consecutive day) daily volatility out-of-sample forecasts when the actual (non-consecutive day) daily RVs are very high. This is different from the finding in Bucci (2020), which studied monthly S&P 500 volatility and found good forecasting performance from ML models (including LSTM) during the Great Recession high volatility period from September 2007 to June 2009, which may include many days when the index volatility was at a 'normal' level. Our sample of individual stocks will have more idiosyncratic volatility that does not persist and is more challenging to forecast.

The finding of OB-ML superior performance in Table 7 contradicts Rahimikia and Poon (2022), which showed news variables to have a stronger predictive power than LOB variables. The contradictory finding could be due to the fact that the ML model here has a large number of predictors, whereas in Rahimikia and Poon (2022), 10 LOB variables and 9 news sentiment measures were added, only one at a time, to the CHAR model. In contrast, News/OB-ML, the largest ML model tested here, has 147 predictors, i.e., 132 LOB variables, 9 news variables and 6 HAR variables, all included and tested in a single ML model. Hence, apart from the ability to capture non-linear relationships as highlighted in Christensen et al. (2022), the ability of the ML model to fit a large number of predictors jointly significantly amplifies its predictive accuracy, particularly on normal volatility days. At the time of writing, this study tested the largest number of predictors (and their combinations) in the ML models for RV forecasting.

In Figure 3, the white (hatched) bar reports the RC values for a specific ML model (CHARx) against all HAR-family of models. The dashed (solid) line represents the out-of-sample average (median) $\Delta_{MSE,j}$ in panel (a), and average (median) $\Delta_{QLIKE,j}$ in panel (b). The news variables in the CHARx model (represented by hatched bars) are defined in Subsection 3.2. The LOB variables include the *type 1 modified slope* (Næs and Skjeltorp, 2006), the *type 2 slope* (Kalay et al., 2004), and the *LOB depth*. A variable name, $T1slope(5bid)$, means *type 1 slope* aggregated from the first five levels on the bid side of the LOB. If a LOB variable name does not contain 'ask' or 'bid', it means that it is calculated using both the bid and ask sides of the LOB.

The most striking result in Figure 3 is the superior forecasting performance of the four ML models (represented by white bars) compared to the CHARx (represented by hatched bars) for normal volatility days in the two upper graphs. In contrast, all four ML models performed poorly on high volatility days in the two lower graphs. News count and LOB depth are the only variables that, when added to the CHAR model, help to forecast RV on high volatility days. The results for MDA presented in Figure C1 of Appendix C broadly support these findings here, viz. a significant improvement in RV forecasting performance on normal volatility days when switching from CHARx to ML, but significant degradation in performance on high volatility

days. These findings from the MDA provide evidence that the ML models are learning the dynamics of RV properly, and the possibility of over-fitting is relatively low.

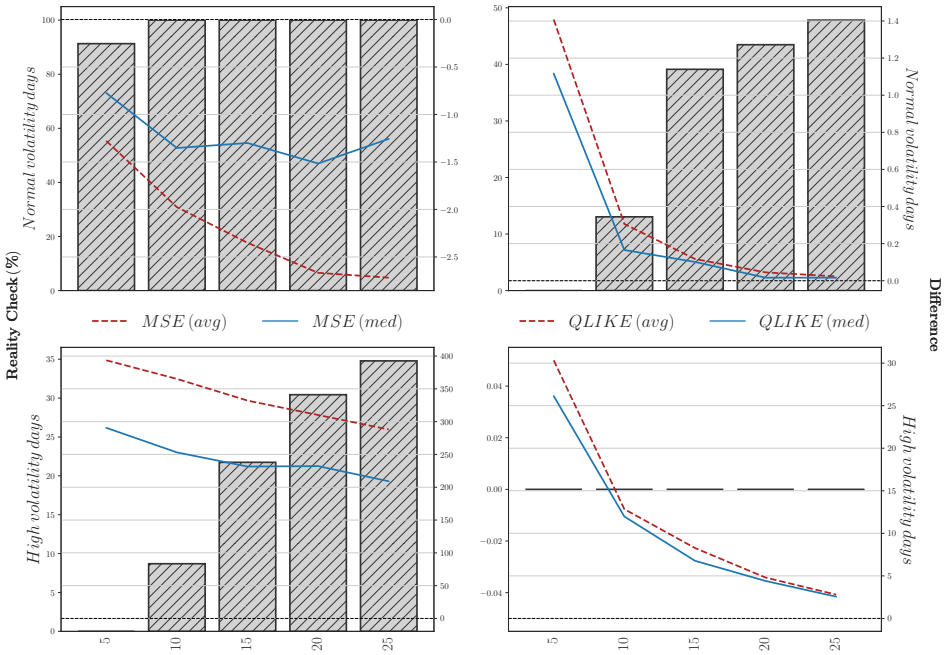
5.3 ML with Limited Feature Set

In this subsection, only past values of RV are used. The information set is restricted to 21 lags of RV, including its previous day value, the average of last week, and the average of last month. The results are presented in Figure 4, where $\Phi_t = \{RV_t, \ ..., RV_{t-20}\}$ for the LSTM model in Figure 4a, and $\Phi_t = \{RV_t, \overline{RV}^w_{t-1}, \overline{RV}^m_{t-1}\}$ for the simpler FCNN model in Figure 4b.[1] FCNN with three input variables is equivalent to the HAR model structure but with a potentially non-linear relationship. Within each subfigure, MSE (QLIKE) is on the left (right), and normal (high) volatility days is at the top (bottom).

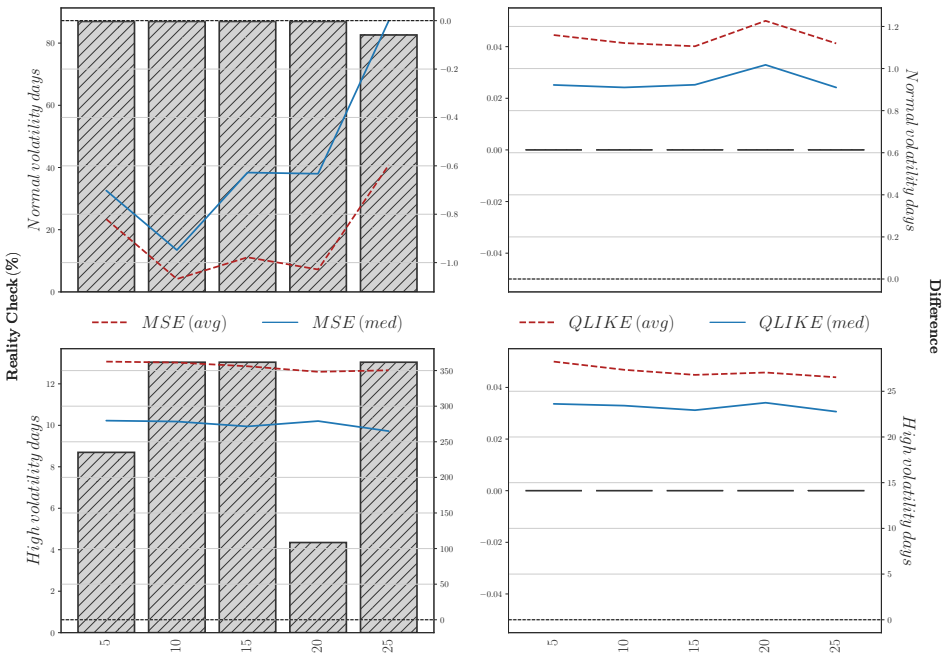
In the top-left (North-West corner) of both Figure 4a and Figure 4b under MSE for normal volatility days, the ML model, generally with a higher number of units, continues to outperform CHAR and HAR-family of models. The average and median Δ_{MSE} are negative in all cases. The RC values are also close to 100% in many cases. As before, the ML models perform poorly on high volatility days. However, results under QLIKE are less marked compared to those under MSE, especially for FCNN (as compared to LSTM). Furthermore, a comparison between Figure 4 and Figure 3 indicates clearly that limiting the predictors to include only the history of RV and HAR variables, significantly degrades forecasting performance. This deterioration is evident not only in contrast to models with a greater number of predictors such as OB-ML and News/OB-ML, but also when compared to the smaller scale HAR-ML model. Hence, consistent with the outcomes presented in Subsection 5.2, apart from nonlinearity, a diverse range of data features plays a crucial role in the ML model achieving stella RV forecasting performance. The MDA results presented in Appendix D (Figure D2 for LSTM and Figure D3 for FCNN) produce findings consistent with the discussion here.[2]

[1]The FCNN has a simple structure, unlike the LSTM. It consists of only three layers, viz. input, hidden and output layers. The size of the input layer is the same as the number of input variables. The hidden layer size varies from 5, 10, 15, 20, to 25. The size of the output layer is equal to the number of output(s), which is one in this study. The dropout rate between the hidden layer and the output layer is set equal to 0.5. The activation functions are sigmoid and rectifier, respectively, for the hidden and the output layers. The other ML model specifications remained unchanged.

[2]ML provides a substantial MDA improvement over the HAR-family of models for normal volatility days. In some cases, the improvement reaches over 4%, and the RC values are as high as near 100%. Nevertheless, the ML performance is poor on high volatility days. Furthermore, it is evident that a diverse range of input variables significantly contributes to ML model achieving a notable increase of over 7% in MDA.



(a) LSTM with $\Phi_t = \{RV_t, ..., RV_{t-20}\}$



(b) FCNN with $\Phi_t = \{RV_t, \overline{RV}^w_{t-1}, \overline{RV}^m_{t-1}\}$

Figure 4: ML models with 21-day information set $\Phi_t = \{RV_t, ..., RV_{t-20}; RV_t, \overline{RV}^w_{t-1}, \overline{RV}^m_{t-1}\}$

Notes: The bar chart is the percentage of tickers with outstanding performance considering the MSE (left figures) and QLIKE (right figures) loss functions at the 5% significance level of the RC compared to the HAR-family of models as the benchmark for every specified number of units of the ML model. The values for the bar chart can be read from the left-hand axis. The dashed (solid) line represents the difference between the average (median) of the out-of-sample MSEs and QLIKEs of the specified ML models and the CHAR model for 23 tickers (negative value shows improvement, and positive value shows degradation in performance). The values for the solid and dashed lines can be read from the right-hand axis. The horizontal dashed line represents no improvement.

5.4 Individual Stocks Radar Plots

To understand the changes in forecasting performance at the ticker and model levels, this subsection analyses the $\Delta_{MSE,i}$ and $\Delta_{QLIKE,i}$ between ML model (OB-ML with 15 units as the best performing ML specification in Table 7) and each of the HAR-family of models[1] using radar plots in Figure 5. The MSE (QLIKE) results are presented on the left (right), and the normal (high) volatility days results are at the top (bottom). A negative Δ value shows the ML model outperformed, and a positive Δ indicates the ML model underperformed. The bold inner circle represents $\Delta = 0$, i.e., there is no difference in performance between the two types of models.

For normal volatility days in Figure 5a, it is apparent that OB-ML outperformed every model in the HAR-family of models as $\Delta_{MSE,i}$ and $\Delta_{QLIKE,i}$ are mostly negative and lie inside the bold circle for most of the tickers. For high volatility days presented in Figure 5b at the bottom, most $\Delta_{MSE,i}$ and $\Delta_{QLIKE,i}$ are positive and lie outside the bold circle, i.e., ML underperformed all HAR-family of models. The $\Delta_{MSE,i}$ and $\Delta_{QLIKE,i}$ are extremely large for six stocks, viz. 'NVDA', 'AMAT', 'ADSK', 'MU', 'NTAP' and 'KLAC'. From the descriptive statistics of RV in Table 1, RV of 'MU' has a very high standard deviation, while RV of 'NVDA', 'AMAT', 'ADSK', 'NTAP' and 'KLAC' has a very high skewness. The MDA results presented in Figure C2 are consistent with those from MSE and QLIKE above.[2] These results confirm the findings in Subsection 5.2 that the ML model dominates the HAR-family of models on normal volatility days but underperforms on high volatility days, especially for stocks exhibiting extreme volatility distribution. Moreover, consistent with the findings in Rahimikia and Poon (2022), the red dotted line, representing CHAR, lies closest to the bold circle suggesting that CHAR has the best forecasting performance among the HAR-family of models, albeit not as good as ML. Also, there are some substantial differences in forecasting performance of Har-family of models both for normal and high volatitliy days some of them showing much worse perforamnce than CHAR. This can be considered as one of the reasons why RC values

Moreover, it's clear that although CHAR model forecating performance

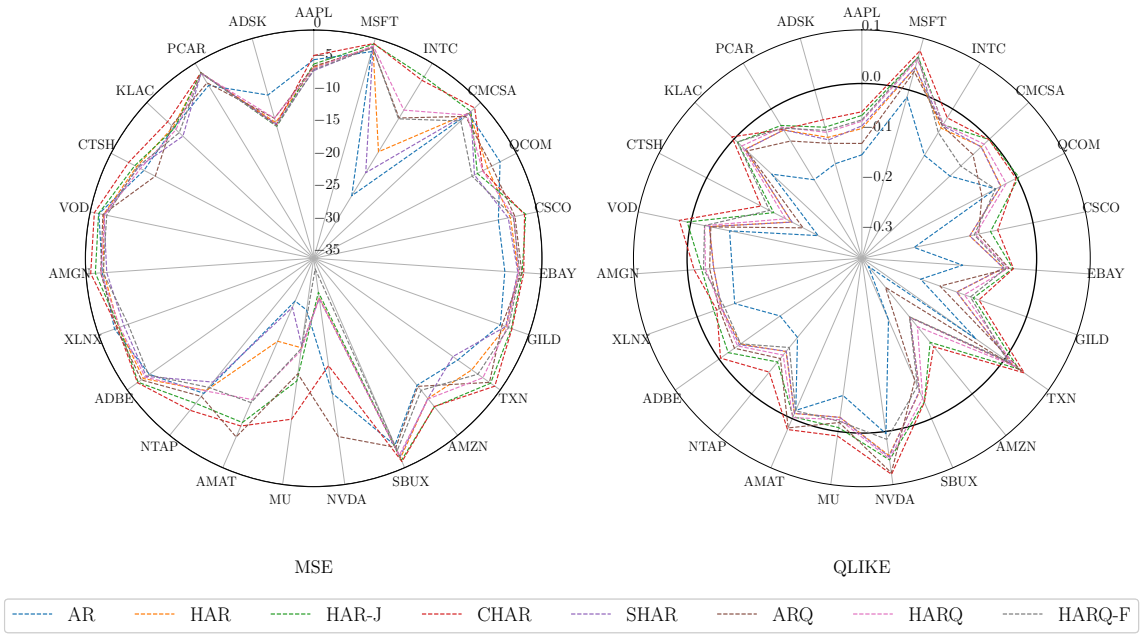
Figure 6 uses a box chart to present the distribution of true RV and the forecast RVs from the OB-ML model with 15 and 25 units and CHAR, respectively. For the sake of clarity, RV on the y-axis is truncated at 100%. The points above each box chart are high volatility days.

[1] That is AR1, HAR, HAR-J, CHAR, SHAR, ARQ, HARQ, and HARQ-F.

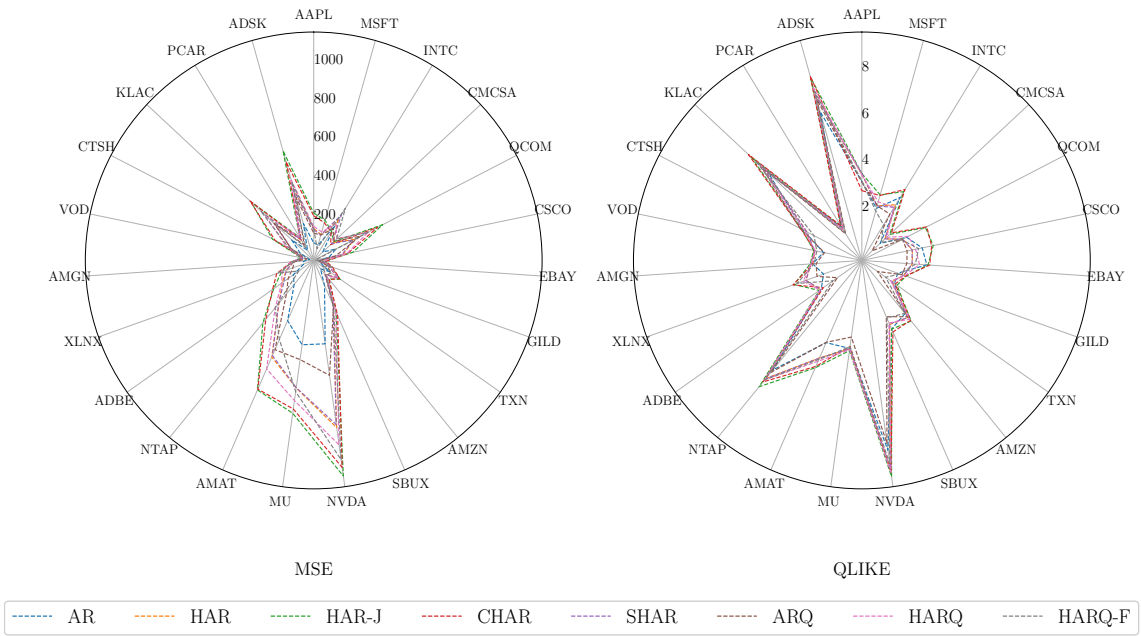
[2] In Figure C2, a positive MDA value indicates ML model outperforms HARs, and the reverse is true for a negative MDA value. Consistent with previous results, for normal volatility days on the left, all the Δ_{MDA} lie outside the bold circle, where ML outperformed every HAR-family model substantially. In some cases, this improvement is nearly 10%. In contrast, for high volatility days on the right, ML underperformed the HAR-family of models.

There are some important observations from Figure 6. CHAR produced some forecasts that are classified as high volatility days. None of the ML models produced volatility forecasts that can be classified as high volatility. CHAR produced more conservative RV forecasts compared with actual RV. The ML model produced RV forecasts that are even more conservative, capped at about 20%. Hence, they both performed poorly on high volatility days. Closer inspections of the other HAR-family of models revealed the same pattern. In general, the HAR-family of models produced forecasts that have higher means and higher standard deviations in the out-of-sample period than those from the ML models.[1] This is one of the reasons for the very poor ML model forecasting performance on high volatility days. Increasing the number of units in the ML model increases the mean and standard deviation of the RV forecasts, resulting in better forecasting performance on high volatility days.

[1]The forecasts from AR1, HAR, HAR-J, CHAR, SHAR, ARQ, HARQ, and HARQ-F are higher than the true RV, respectively, 77.6%, 78.7%, 79.8%, 80.2%, 78.6%, 78.0%, 78.8%, and 77.6% of time (average across 23 stocks), while the forecasts from OB-DL with 15 and 25 units are higher than the true RV, 59.5% and 65.0% of the time respectively. The standard deviations of the forecasts from the eight HAR-family of models are higher than the standard deviation of the forecast from OB-DL with 15 units (25 units) for, respectively, 91.3%, 100%, 100%, 100%, 100%, 100%, 100%, and 100% (82.6%, 100%, 100%, 100%, 100%, 100%, 100%, and 100%) of the 23 tickers.



(a) Normal volatility days



(b) High volatility days

Figure 5: $\Delta_{MSE,i}$ and $\Delta_{QLIKE,i}$ between OB-ML (15 units) and HAR-family of models

Notes: Every radar chart contains the difference between the performance of the OB-ML model with 15 units with the AR1, HAR, HAR-J, CHAR, SHAR, ARQ, HARQ, and HARQ-F models for the mentioned tickers considering normal volatility days in Figure 5a and high volatility days in Figure 5b. The left and right radar charts contain results for the MSE and QLIKE loss functions, respectively. For these loss functions, the negative value shows improvement. The bold circle inside these radar charts shows no improvement (zero).

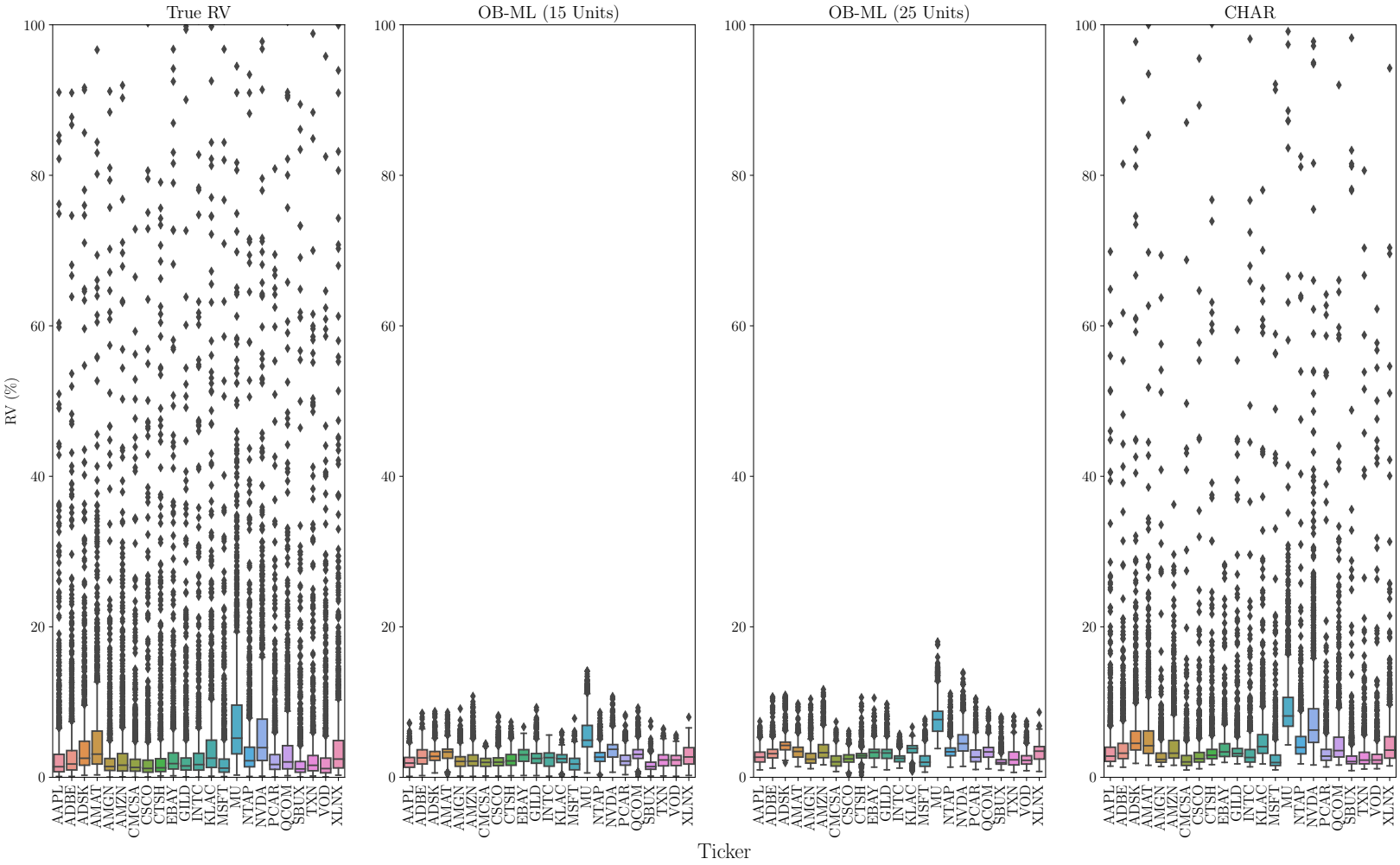


Figure 6: True RV and forecast RVs from OB-ML and CHAR

Notes: From left to right, box charts show the distribution of the true RV, the forecast RVs from OB-ML models with 15 and 25 units, and CHAR, respectively. For the sake of clarity, RV at the y-axis is truncated at 100%. The points above each box chart are high volatility days.

6 Explainable AI

In this section, we calculate the SHAP (SHapley Additive exPlanations) values for the set of 147 (i.e., six HAR, nine news, and 132 LOB) variables included in the News/OB-ML model (with 25 units). Instead of the best performing ML model, OB-ML, we use News/OB-ML because News/OB-ML contains all variables and the goal here is to examine the explanatory power of every variable in the 147 variables set.[1] Lundberg and Lee (2017) introduced SHAP as a type of sensitivity analysis within the realm of *Explainable AI* techniques. Shapley value ϕ_i, defined below, measures the importance of variable i:

$$\phi_i = \frac{1}{|N|!} \sum_{S \subseteq N \setminus \{i\}} |S|! \ (|N| - |S| - 1)! \ [f(S \cup \{i\}) - f(S)], \tag{27}$$

where $N = 147$ is the total number of variables included in the News/OB-ML model, S represents all possible subsets from $N \setminus \{i\}$ (i.e., the set of 146 variables after excluding i^{th} variable), $f(S \cup \{i\}) - f(S)$ is the increase in RV forecast by adding variable i to the set S, and $|N|$ is the number of non-zero entries in N. $|S|!$ and $(|N| - |S| - 1)!$ show the number of ways the chosen and the remaining set of variables may be presented. The resulting value is the average contribution to RV forecasts of input variable i^{th}. In this study, we utilise a high-speed approximation algorithm called Deep SHAP based on DeepLIFT (Shrikumar et al., 2017). For each of the 23 stocks, we calculate SHAP for every day in the out-of-sample period. A variable is among the most important variables for each stock when it is among the top 10% (i.e. 15 out of 147 variables), and also it appeared in the list at least 50% of the times (i.e. 802 out of 1604 days) in the out-of-sample period.

The consolidated results across 23 stocks are reported in Table 8, displaying the number of times the variable is chosen as an important variable. The findings clearly indicate that primary financial variables, including 'Mid prices' (at various LOB levels from 1 to 10), 'Mean bid', and 'Mean ask', are among the most important variables. Table 8 is also dominated by LOB variables. Only three non-LOB variables are included in Table 8, viz. BPV from the list of HAR variables,[2] and 'News count' and 'Uncertainty' from the set of news variables.[3] Finally, many complex LOB-derived variables are chosen as important variables but only for a few stocks suggesting idiosyncratic behaviour among individual stocks.

[1] From Table 7, News/OB-ML with 25 units shows relatively better forecasting performance in the News/OB-ML group considering all loss functions.

[2] Rahimikia and Poon (2022) also reported that the CHAR model, with BPV as the predictive variable, is the best performing HAR model for forecasting RV.

[3] Similarly, Rahimikia and Poon (2022) found, among all the news variables, 'News Count' is the most powerful for forecasting RV.

Table 8: Variable aggregate importance across 23 stocks

Mid price (L1)	13[a]	Relative intensity (S/bid)	3	Relative intensity (D/ask)	1
Mid price (L7)	11	News count	2	Relative intensity(C/ask)	1
Mean price (bid)	10	Relative intensity$_{\mathrm{V}}$ (E/ask)	2	Relative intensity$_{\mathrm{V}}$ (E/bid)	1
Mean price (ask)	9	Mean volume (bid)	2	Bid-ask spread (L2)	1
Mid price (L4)	9	Mean volume (ask)	2	Average intensity$_{\mathrm{V}}$ (E/ask)	1
Mid price (L6)	8	Bipower variation (BPV)	2	Price derivative (L7/bid)	1
Mid price (L10)	8	Average intensity (S/ask)	2	Acceleration$_{\mathrm{V}}$ (E/bid)	1
Mid price (L3)	8	Average intensity (C/ask)	2	Price derivative (L9/bid)	1
Mid price (L9)	7	Price derivative (L4/ask)	2	Volume derivative (L4/bid)	1
Mid price (L8)	7	Price derivative (L2/bid)	2	Volume derivative (L6/bid)	1
Price derivative (L6/ask)	6	Price derivative (L10/bid)	2	Price difference (L2/ask)	1
Mid price (L2)	6	Volume derivative (L10/bid)	2	Price difference (L6/bid)	1
Mid price (L5)	5	Price derivative (L6/bid)	2	Average intensity (C/bid)	1
Average intensity$_{\mathrm{H}}$ (E/bid)	4	Price derivative (L3/ask)	2	Average intensity (D/bid)	1
Price derivative (L2/ask)	4	Price derivative (L5/ask)	1	Acceleration$_{\mathrm{H}}$ (E/ask)	1
Price derivative (L3/bid)	3	Volume derivative (L3/ask)	1	Acceleration (S/bid)	1
Relative intensity (S/ask)	3	Price derivative (L5/bid)	1	Average intensity (S/bid)	1
Relative intensity (C/bid)	3	Price derivative (L1/bid)	1	Acceleration$_{\mathrm{H}}$ (E/bid)	1
Relative intensity (D/bid)	3	Relative intensity$_{\mathrm{H}}$ (E/ask)	1	Uncertainty sentiment	1

Notes: 'L' stands for the level of the LOB. 'E', 'S', 'C', and 'D' stand for execution, submission, cancellation, and deletion of orders, respectively. Also, for execution, 'V' and 'H' stand for visible and hidden execution.
[a] The number of times the indicated variable is chosen as an important variable among 23 tickers.

Next, we made an information set comparison in terms of the group's forecasting performance in the out-of-sample forecasting period. The top chart in Figure 7 shows the daily SHAP average importance (across 23 stocks) of HAR, LOB, and News variable groups. The bottom chart shows the daily RVs of NASDAQ-100 (with 5-min sampling interval) downloaded from Oxford-Man Institute of Quantitative Finance Realized Library.[1] What stands out from Figure 7 is the relatively stable importance of the LOB group over the full out-of-sample forecasting period. HAR variables had the worst forecasting performance overall except during the sporadic, highly volatile episodes. HAR's dominating forecast power during the COVID-19 disruption in 2019 and the extremely high volatility period in early 2020 is very prominent.

The out-of-sample period can be separated into a pre-2018 low-volatility period and a post-2018 period populated by many episodes of fluctuating volatility. Before 2018, the forecasting performance of all three information sets was high but gradually declined as time approached 2018. The forecasting power of news sentiment dominated LOB and HAR variables before 2018, when the market was calm. Post-2018, the forecasting power of news sentiment steadily declined and was dominated by LOB variables. There were a few short time intervals when news sentiment variables forecasting power dominated, but these did not coincide with the timing of the most volatile periods. In summary, it is clear that (i) the predictive power of the three information sets varied through time, and (ii) some predictors are more informative

[1] A note of caution is due here since there are some missing daily RVs in this library.

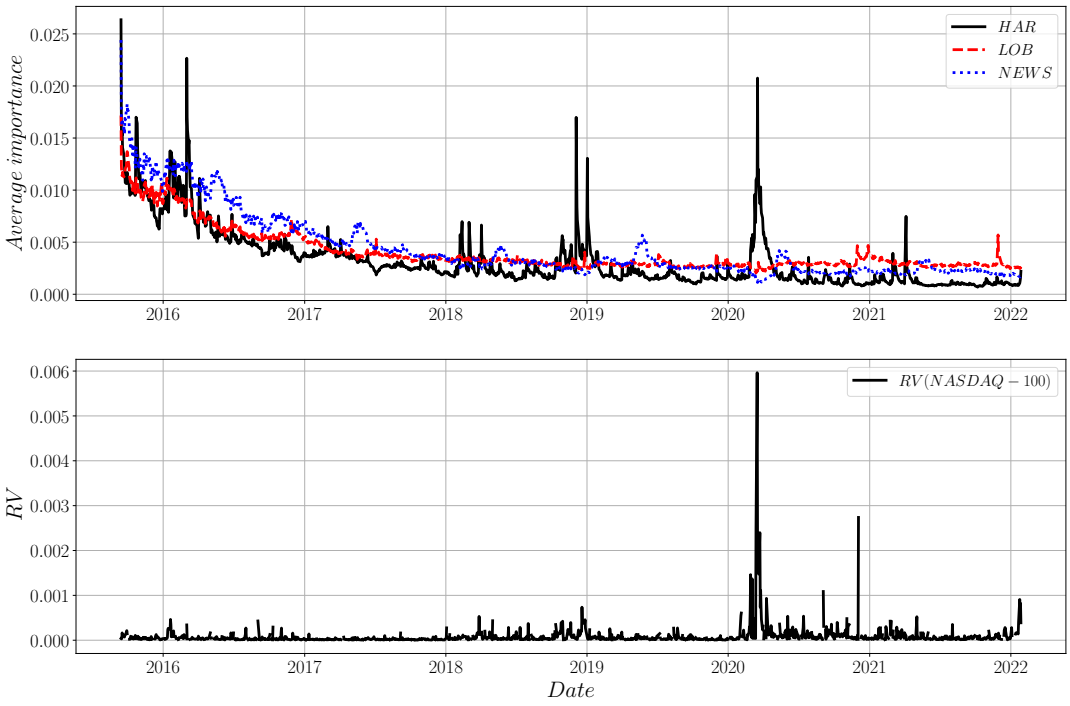


Figure 7: Average variable importance vs market RV (out-of-sample)

Notes: The top chart shows the daily SHAP average importance (across 23 stocks) of the HAR variable group in Subsection 3.1, LOB variable group in Subsection 3.2, and News variable group in Subsection 3.3. This analysis uses News/OB-ML (25 units) to cover all variables together. The bottom chart shows the daily RVs of NASDAQ-100 (5-min) downloaded from the Oxford-Man Institute of Quantitative Finance Realized Library.

during extremely volatile periods, while other predictors are more informative during normal volatility periods. The 1-step ahead re-estimated and re-forecast, flexible ML structure (where there is no fixed time chain, and past information is flexibly controlled by the LSTM gates at every time step), can accommodate the complex patterns in (i) and (ii). Also, according to Section 5, the variation in significant predictors over time reinforces the pivotal importance of flexibility of the ML model in integrating a diverse and wide range of predictive variables for RV forecasting.

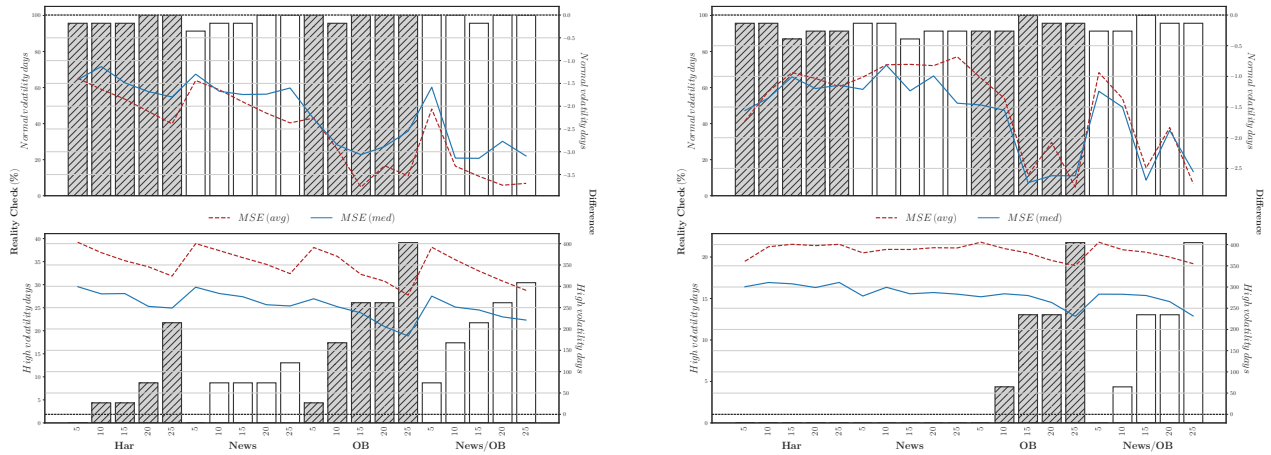
7 Robustness Checks

Up to now, all results in Section 5 show that the ML model with LOB variables outperformed the HAR-family of models for forecasting RV on normal volatility days but not on high volatility days. In this section, we perform a series of robustness checks for this conclusion by using time-restricted input information in Subsection 7.1, using QLIKE (instead of MSE) as the loss function in Subsection 7.2, and a large combination of tuning parameters in Subsection 7.3 and Subsection 7.4. Apart from the experimented parameters in each subsection, all the other model specifications are the same as those adopted in Subsection 4.4.

(a) Forecasts evaluated under MSE

(i) 5 Lags

(ii) 1 Lag



(b) Forecasts evaluated under QLIKE

(i) 5 Lags

(ii) 1 Lag

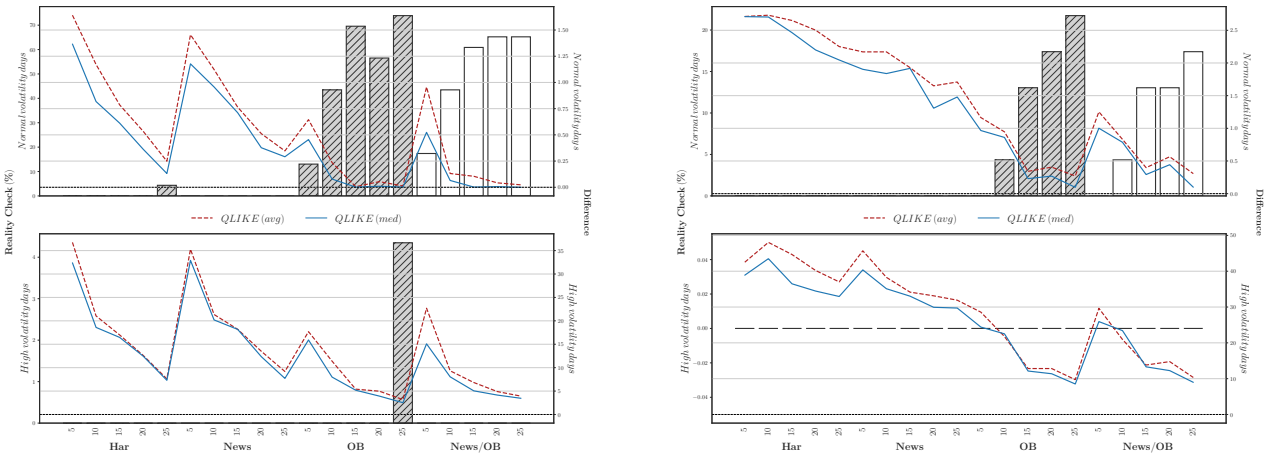


Figure 8: ML models with restricted information set $\Phi_t = \{t, \ ..., t-4\}$

Notes: The left and right figures display the results considering five lags (last week) and one lag (last day). The bar chart is the percentage of tickers with the outstanding performance considering the MSE/QLIKE loss function at the 5% significance level of the RC compared to the HAR-family of models as the benchmark for every HAR-ML, News-ML, OB-ML, and News/OB-ML model. The values for the bar chart can be read from the left-hand axis. The dashed (solid) line represents the difference between the average (median) of the out-of-sample MSEs and QLIKEs of the HAR-ML, News-ML, OB-ML, and News/OB-ML models and the CHAR model for 23 tickers (negative value shows improvement, and positive value shows degradation in performance). The values for the dashed and solid lines can be read from the right-hand axis. The horizontal dashed line represents no improvement.

(a) Normal volatility

(b) High volatility

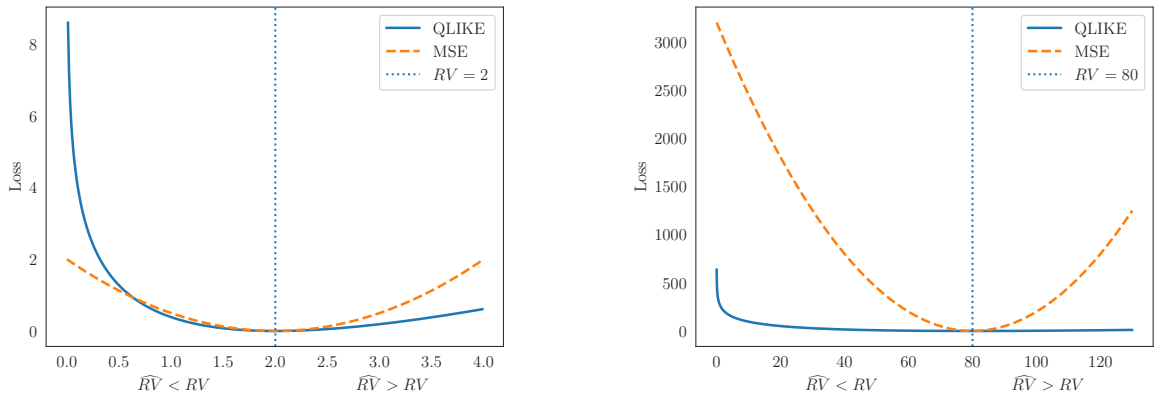


Figure 9: A representation of MSE and QLIKE loss functions

Notes: This graph presents the shape of the MSE and QLIKE loss functions. In this example, the true RV (vertical dashed line) is equal to 2 for (a) Normal volatility or 80 for (b) High volatility. To the left of the true RV, $\widehat{RV} < RV$, and to the right of the true RV, $\widehat{RV} > RV$.

7.1 Restricted Number of Lags, $\Phi_t = \{t, \ ..., t-4\}$

One may argue that the superior ML performance is solely due to the large amount of information it uses. Here, we restrict all input variables to have five lags (equivalent to one week) and one lag (i.e. the previous day). An LSTM with one lag (one input and one output) reduces to a standard neural network. Figure 8 presents the average (as a solid line) and median (as a dotted line) Δ_{MSE} and Δ_{QLIKE}, as well as the RC values (as a bar chart). The results show that restricting the information set from 21 days in Figure 3 to 5-days and 1-day did not change the superior performance of the ML models for normal volatility days when evaluated using MSE. Also, the QLIKE results are weaker but show a noticeable improvement in forecasting performance. The 1-day results are also weaker than the 5-day results (see the top half of Figure 8a for 5-days MSE in (i) and for 1-day MSE in (ii) and top half of Figure 8b for 5-days QLIKE in (i) and for 1-day QLIKE in (ii)). As before, the ML models are not suited for forecasting RV on high volatility days. The results based on MSE in the bottom half of Figure 8a and the results based on QLIKE in the bottom half of Figure 8b all support this conclusion. The MDA results presented in Figure D1 of Appendix D are also consistent with the findings here.[1] Here, we conclude that our findings of superior ML forecasting performance on normal volatility days are not due to the larger historical information set that it has access to but strictly due to its more flexible model structure.

7.2 MSE vs QLIKE as Loss Function

All the ML model results produced so far are based on minimising MSE as the loss function in the training period while using MSE and QLIKE in forecast evaluation. Here, we test if changing the loss function to minimising QLIKE will change the results and conclusions. According to Patton (2011), MSE and QLIKE are members of a family of robust and homogeneous loss functions, $L(\cdot)$, below:

$$L(RV, \widehat{RV}; b) = \begin{cases} \frac{1}{(b+1)(b+2)}(RV^{b+2} - \widehat{RV}^{b+2}) - \frac{1}{b+1}h^{b+1}(RV - \widehat{RV}), & \text{for } b \notin \{-1, -2\} \\ \widehat{RV} - RV + RV(\log(\frac{RV}{\widehat{RV}})), & \text{for } b = -1 \\ \frac{RV}{\widehat{RV}} - \log(\frac{RV}{\widehat{RV}}) - 1, & \text{for } b = -2, \end{cases} \tag{28}$$

where L is the loss function, RV is the true RV, $\widehat{RV}$ is the fitted (forecasted) RV, and b is the scalar parameter. For $b = 0$, L becomes MSE , and L becomes QLIKE if $b = -2$. Figure 9 presents the shape of these two loss functions setting the true RV (vertical dashed line) equal to 2 (in Figure 9a) and 80 (in Figure 9b). To the left (right) of the true RV, $\widehat{RV} < RV$ ($\widehat{RV} > RV$). Compared to MSE, QLIKE is asymmetric and penalises large under-forecasts more than large over-forecasts. Patton and Sheppard (2015) found, in volatility forecasting, that the power of DMW tests (Diebold and Mariano (1995) and West (1996)) are higher when the loss function is QLIKE instead of MSE, suggesting that QLIKE might be a better loss function for ranking competing volatility forecasting models.

As a robustness check, we compare MSE vs QLIKE as the loss function for training the ML models, while all other model specifications remain unchanged. QLIKE (as in Equation (19) and Equation (28)) is undefined when $\widehat{RV} \leq 0$. To avoid this limitation, we add a lambda layer as the last layer after the FCNN, making the $\widehat{RV} \geq 0.01$, i.e. always greater than zero.[1] The results are presented in Figure 10. Figure 10a (Figure 10b) evaluates the forecasts using MSE (QLIKE), while the left (right) figure within each subfigure shows the results for the ML models trained using minimising MSE (QLIKE) as the loss function.

As before, there are huge gains from ML models, improving volatility forecasts on normal days but a performance degradation on high volatility days. Figure 10a shows that changing

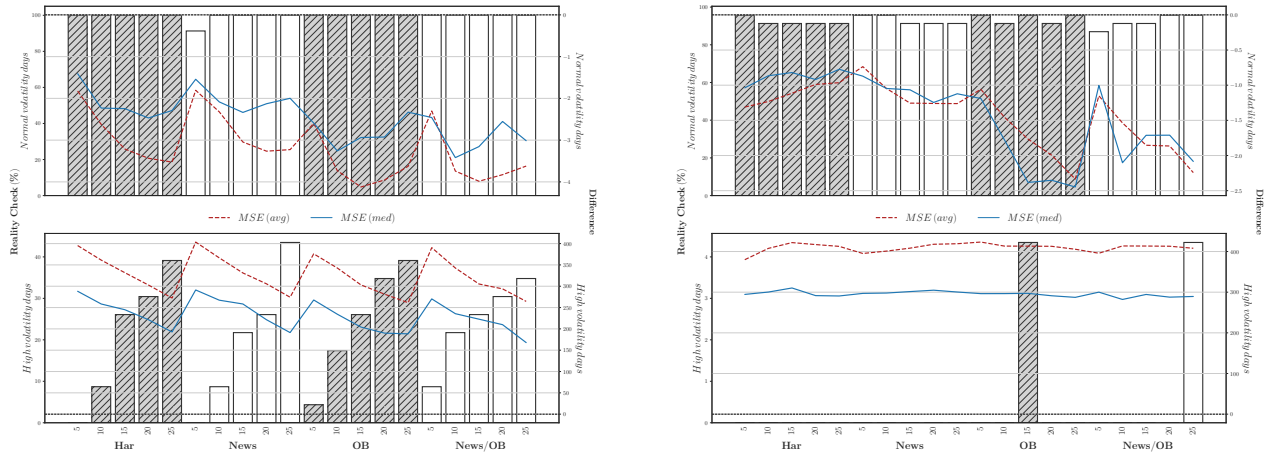
[1]For normal volatility days, the RC values of ML models are high, and many reach 100%. The improvement in MDA by switching from CHAR to ML varies between 1% to 7% for ML with the different number of units and the different number of lagged variables. In summary, these results show that ML outperforms the HAR-family of models in RV forecasting for normal volatility days, even with only 1 and 5 lags instead of 21 lags.

[1]Another way of making sure $\widehat{RV} > 0$ is to change the scalar parameter, b, to very close to -2 like -1.99 or -2.01 in Equation (28). In order to keep consistency with the other experiments, this method is not utilised in this study.

(a) Forecasts evaluated under MSE

(i) MSE loss function

(ii) QLIKE loss function



(b) Forecasts evaluated under QLIKE

(i) MSE loss function

(ii) QLIKE loss function

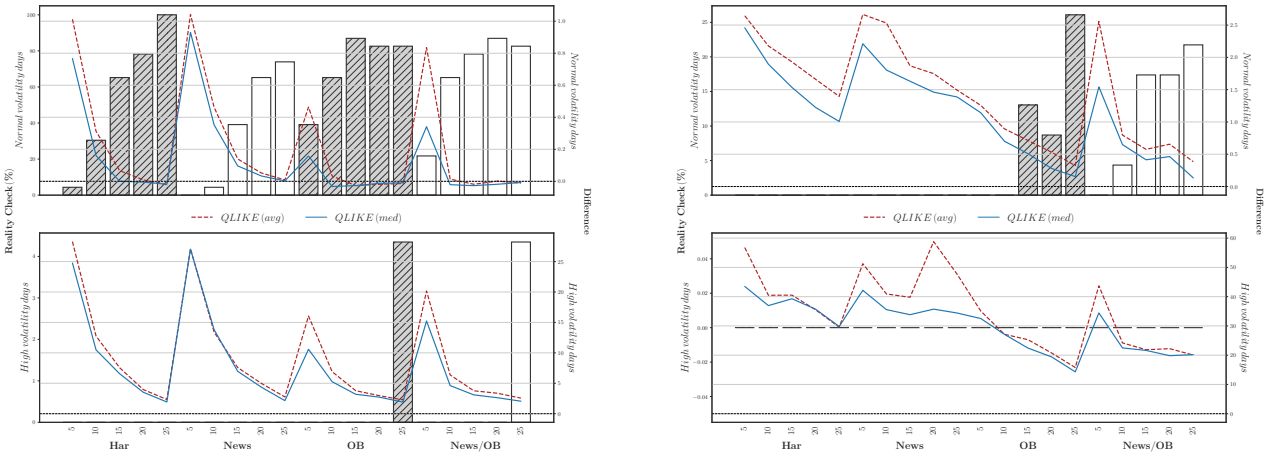


Figure 10: Minimising MSE vs QLIKE as the loss function in training period

Notes: For each of the two subfigures, the left (right) figure presents the results for the four ML models (HAR-ML, News-ML, OB-ML, and News/OB-ML) trained using minimising MSE (QLIKE) as the loss function. The other model specifications are the same as in Section 3. The bar chart is the RC percentage of tickers with outstanding ML performance at the 5% significance level against all the HAR-family of models. The values for the bar chart can be read from the left-hand axis. The red dashed (blue solid) line represents the average (median) $\Delta_{MSE/QLIKE}$ for each of the four ML models (HAR-ML, News-ML, OB-ML, and News/OB-ML) against CHAR for 23 tickers; a negative value indicates improvement, and a positive value indicates a performance degradation. The values for the dashed and solid lines can be read from the right-hand axis. The horizontal dashed line represents no difference in performance between ML and CHAR.

MSE to QLIKE loss function causes a substantial degradation in forecasting performance for high volatility days. Considering both RC and the average and median Δ_{MSE} values for normal volatility days, the degradation is evident but not as severe as on high volatility days. Further analysis of forecasts evaluated under the QLIKE loss function in Figure 10b shows a similar but more severe degradation pattern in performance by switching to the QLIKE loss function. It can be seen that the degradation for normal volatility days is high; many RC values are near zero, and all the average and median Δ_{QLIKE} values are positive and large, showing a substantial degradation in forecasting performance. Moving to high volatility days, incorporating the QLIKE loss function in the training process changes all RC values to zero with substantial degradation in Δ_{QLIKE} values. Results for the MDA loss function are also in line with these conclusions.[1]

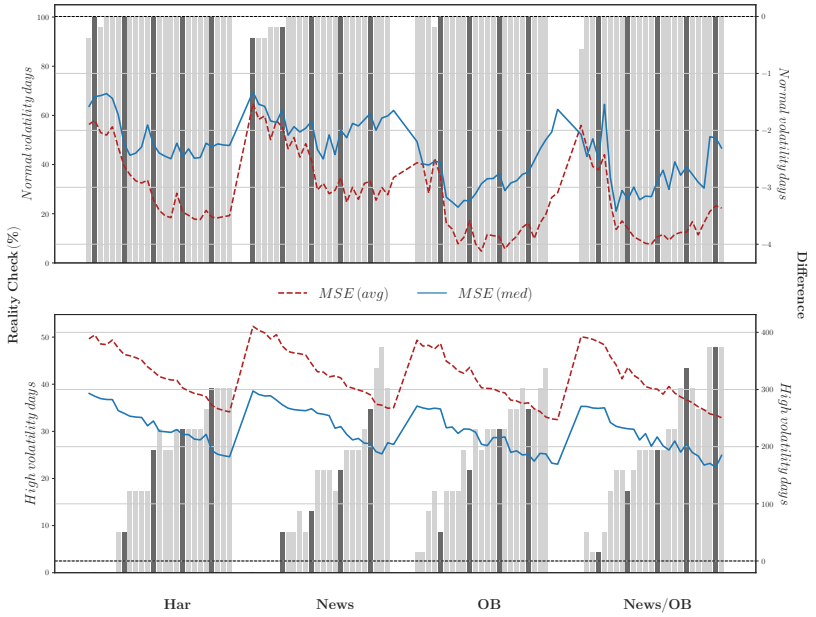
However, what remains unclear is why when minimising QLIKE (instead of MSE) as the loss function, the ML models severely under-perform when QLIKE is designed to avoid large under-forecasts. The clue lies in the QLIKE weighting function when true RV is very high. Returning to Figure 9 where the MSE and QLIKE loss functions are presented for true $RV = 2$ on the left and true $RV = 80$ on the right, it is clear that when the true RV is very high, the weights of QLIKE become very flat on both sides of the true RV; only when $\widehat{RV} << RV$, the weight begins to rise. This means that when true RV is very high, QLIKE becomes insensitive to the size of the forecast errors (except for extremely big under-forecasts). Hence, the optimisation algorithm loses the ability to learn to forecast accurately. This evidence suggests that QLIKE, without further modification, is not appropriate as a loss function for training ML models, at least as specified here. Therefore, only MSE is used as the loss function in this study.[1]

7.3 No of Units vs No of Epochs

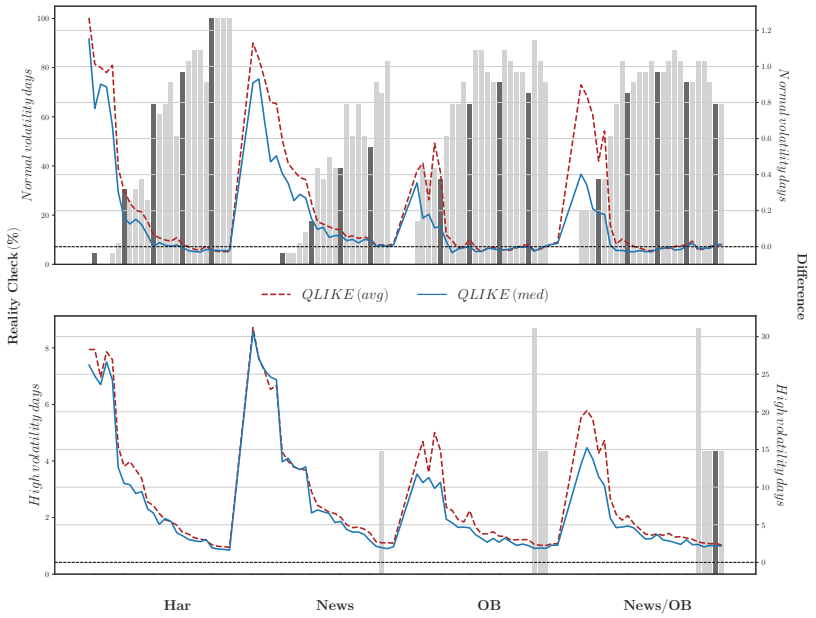
This subsection tests the sensitivity of the number of units (#units) and the number of epochs (#epochs) in affecting the forecasting performance of ML models on normal volatility days as well as on high volatility days. We test the number of units ranging from 5, 10, 15, 20, to 25, and for the number of epochs, from 25, 50, 75, 100, to 125. The other ML model specifications are the same as those in Subsection 4.4. For 23 tickers, four groups of information sets, and 1604 days in the out-of-sample period, 3,689,200 ($25 \times 23 \times 4 \times 1604$) ML models are trained and tested, which is substantially higher than the 737,840 (5 choices of units $\times 23 \times 4 \times 1604$) ML models trained in Section 5.

[1] From Figure D4, changing the MSE loss function in Figure D4a to the QLIKE loss function in Figure D4b causes a noticeable degradation in performance for both normal and high volatility days.

[1] The results here also weaken the use of QLIKE as a measure for forecast evaluation, especially on high volatility days.



(a) Forecasts evaluated under MSE



(b) Forecasts evaluated under QLIKE

Figure 11: No of units vs No of epochs

Notes: From left to right, this figure consists of the HAR-ML, News-ML, OB-ML, and News/OB-ML variable groups. For every group, the results are shown in the following order from left to right (#units-#epochs): 5-25, 5-50, 5-75, 5-100, 5-125, 10-25, 10-50, 10-75, 10-100, 10-125, 15-25, 15-50, 15-75, 15-100, 15-125, 20-25, 20-50, 20-75, 20-100, 20-125, 25-25, 5-50, 25-75, 25-100, and 25-125. For the sake of clarity, these values are not shown in this figure. The bar chart is the percentage of tickers with the outstanding performance considering the MSE loss function in the top part (QLIKE at the bottom) at the 5% significance level of the RC compared to the HAR-family of models as the benchmark for every specified ML model. The darker bar charts are the RC values of primary experiments in Section 5. The RC values can be read from the left-hand axis. The red dashed (blue solid) line represents the average (median) $\Delta_{MSE/QLIKE}$ between ML and CHAR for 23 tickers in the out-of-sample period. A negative value indicates improvement, and a positive value indicates degradation in performance. The values for the dashed and solid lines can be read from the right-hand axis. The horizontal dashed line represents no improvement.

The results are presented in Figure 11a for the MSE evaluation function, and Figure 11b for the QLILE evaluation function, for the four ML models (i.e. from left to right, HAR-ML, News-ML, OB-ML, and News/OB-ML). For each ML tested, the results are shown in the following order from left to right (#units-#epochs): 5-25, 5-50, 5-75, 5-100, 5-125, 10-25, 10-50, 10-75, 10-100, 10-125, 15-25, 15-50, 15-75, 15-100, 15-125, 20-25, 20-50, 20-75, 20-100, 20-125, 25-25, 5-50, 25-75, 25-100, and 25-125. For the sake of clarity, these values are not shown in Figure 11. The lighter bar chart is the percentage of tickers with outstanding performance at the 5% level of the RC. The darker bar charts are also the RC values of primary experiments in Section 5.

First, consider the case of MSE (QLIKE) as the evaluation function in Figure 11a (in Figure 11b). As before, the results confirm that, for normal volatility days, generally, a higher number of units produced the best performance in terms of average (median) $\Delta_{MSE/QLIKE}$ and the RC values. Second, the number of units influences forecasting performance more than the number of epochs. Third, ML outperformed CHAR and HAR-family of models only on normal volatility days; when switching to the high volatility days, ML models under-performed with positive average (median) $\Delta_{MSE/QLIKE}$, and lower than 50% RC values. Also, increasing the number of units and epochs reduces the amount of under-performance on high volatility days. Forth, the results for QLIKE as the evaluation function show the same patterns. Finally, of the four ML models tested here, OB-ML performance is the most outstanding one.[2]

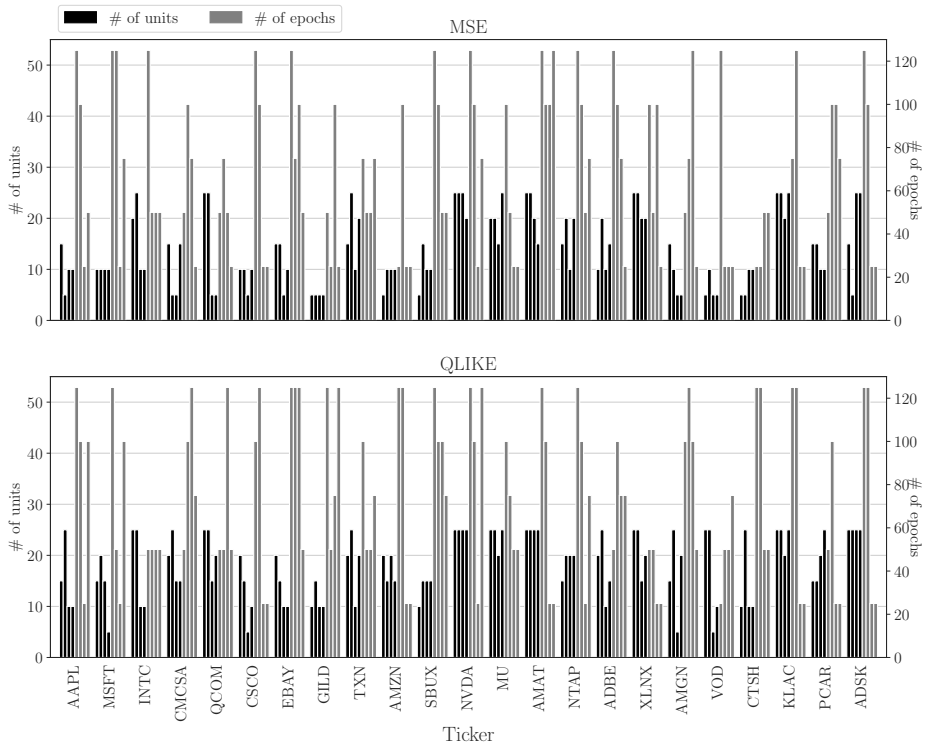
Overall, these results again confirm that ML models, as defined here, exhibit strong volatility forecasting performance for normal volatility days but not for high volatility days. Therefore, more care is needed for forecasting high volatility days. A more complex ML model, with a higher number of units and epochs, has to be trained specifically for forecasting RV on high volatility days.[3]

7.4 Units vs Epochs: Individual Stock Analysis

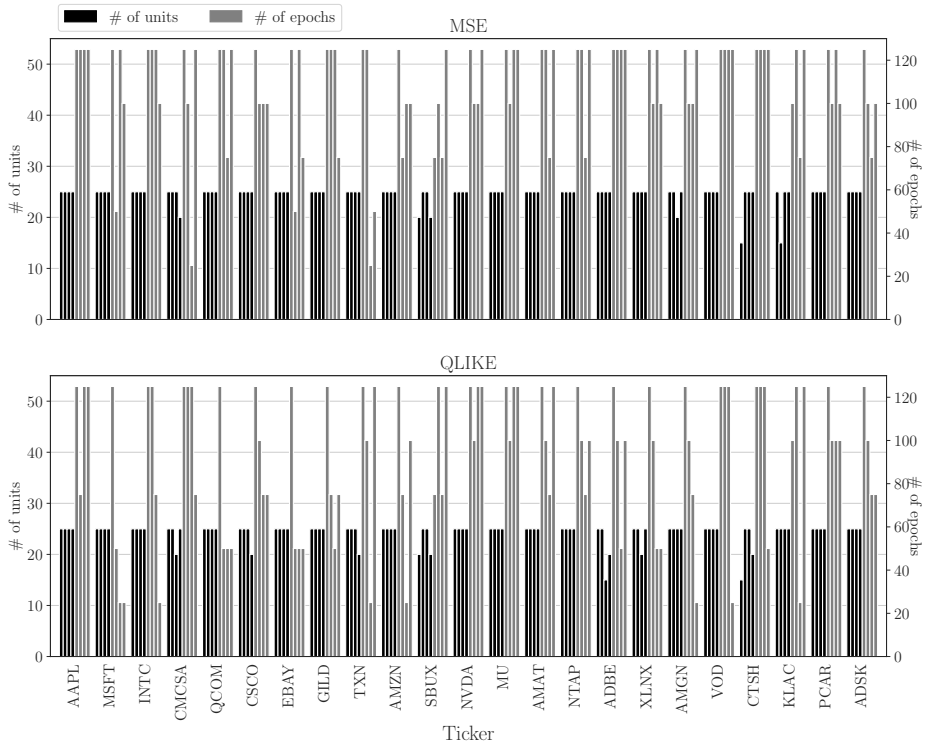
This subsection concerns the best combination of the number of units and epochs for each ticker's best volatility forecasting performance. The results are presented in Figure 12a and Figure 12b for normal and high volatility days, respectively. For each ticker, there are two sets

[2]The results for the MDA in Figure E1 of Appendix E corroborate the findings here. Most RC values for normal volatility days are near 100% and Δ_{MDA} reaching around 8% improvement in the OB-ML group. For high volatility days, poor performance is clear for all combinations of the number of units and epochs.

[3]We deliberately drop the early stopping mechanism in the training process because of three primary considerations. Initially, allocating a segment of the time series for early stopping would lead to excluding the most recent data from the training process. This segment could potentially hold crucial insights for more accurate RV forecasts. Secondly, as detailed earlier, we have tested 3,689,200 distinct models, each employing different combinations of the number of unit and epochs. The findings unanimously point towards a poor performance on high volatility days by all models tested. That is, our conclusion, in general, is not sensitive to the choice of hyperparameters. Thirdly, by dropping early stopping, we can compare performance due entirely to the choice of hyperparameters.



(a) Normal volatility days



(b) High volatility days

Figure 12: The best performing hyperparameters (#units & #epochs)

Notes: There are two sets of four bars for each ticker that correspond to the four ML models, viz. HAR-ML, News-ML, OB-ML, and News/OB-ML. The first four black bars correspond to their optimal number of units, while the next four grey bars correspond to their optimal number of epochs. The top (bottom) half is for normal (high) volatility days. Within each subfigure, the top (bottom) graph reports the results considering the MSE (QLIKE) loss function. The number of units (# of units) can be read from the left axis, and the number of epochs (# of epochs) can be read from the right axis.

of four bars corresponding to the four ML models, viz. from left to right, HAR-ML, News-ML, OB-ML, and News/OB-ML. The first four black bars correspond to the optimal number of units of the four ML models, while the next four grey bars correspond to the optimal number of epochs of the four ML models. The top (bottom) half is for normal (high) volatility days. Within each subfigure, the top (bottom) graph reports the results based on MSE (QLIKE) loss function. The number of units (# of units) can be read from the left axis, and the number of epochs (# of epochs) can be read from the right axis.

The optimal numbers of units and epochs are generally larger for high volatility days than those for normal volatility days. There are also some variations among the tickers, especially for normal volatility days in Figure 12a for both MSE and QLIKE.[4] For high volatility days in Figure 12b, there are fewer variations across tickers, and the results are in favour of a more complex ML model with about 25 units and 70 to 120 or more epochs.

The individual ticker analysis gives important insight into the importance of hyperparameter choice conditioned on the level of actual RV. A more complex ML model specification (with a high number of units and epochs) works well for almost all tickers for forecasting RV on high volatility days. In contrast, if to forecast RV on normal volatility days, one will expect more idiosyncratic variations, and careful tuning of hyperparameters for every input variable group and for each ticker can provide additional improvement to RV forecasting performance.

8 Discussions and Conclusions

This study examines the efficacy of ML models alongside a comprehensive set of features for RV forecasting in 23 NASDAQ stocks spanning from 27 July 2007 to 27 January 2022. Three types of daily data are used: six HAR variables, 132 LOB variables, and nine news sentiment variables. The entire sample period is split into an in-sample training period from 27 July 2007 to 11 September 2015 and an out-of-sample forecasting period from 14 September 2015 to 27 January 2022. Using the LSTM model combined with a FCNN layer and four sets of input variables, each with 21 lags, 3,689,200 variants of ML models are trained and tested in this study.

Our empirical tests provide overwhelming evidence that ML models outperformed all HAR-family of models, and the LOB variables, in comparison to News sentiments and HAR variables, emerged as the most powerful predictors for forecasting RV. Our findings remain qualitatively the same when the forecasts are evaluated using MSE, QLIKE, MDA or RC. However, this

[4]Looking at the top half of Figure 12, it is apparent that for some of the tickers like 'NVDA', 'MU', 'AMAT', and 'ADSK', a more complex model with a higher number of units is required for better forecasting performance, consistent with the findings in Subsection 5.4.

statistically significant improvement applies only to 90% of out-of-sample daily forecasts when actual RV is not extremely high. For a small number of stocks and on 10% of the out-of-sample forecasts when the actual RV is extremely high, HAR-family of models outperformed ML models. Particularly, our findings emphasise not only the importance of capturing nonlinear relationships in the RV dynamics but also the importance of including a rich collection of predictors in a single forecasting ML model.

The neural network algorithm in ML model is a black box, giving very little clue about which predictors contributed to good forecasting performance. Here, we implemented SHAP, an *Explainable AI* technique, to identify the predictors that provided the greatest forecasting power. Results from SHAP suggest, among the 147 input variables, mid prices at all LOB levels, average bid and average ask from LOB, BPV from HAR, news count and 'uncertainty' sentiment (compiled following (Loughran and McDonald, 2011) using Dow Jones News) are identified as the most powerful predictive variables for forecasting RV. In general, LOB variables have stronger predictive power than HAR and News sentiment variables. There are also some performance variations across time, however. The News sentiment variables had strong forecasting power before 2018. HAR's forecast power dominated during the COVID-19 disruption in 2019, and the extremely high volatility period in early 2020. From 2020 onward, LOB variables emerged strongly as the most powerful predictive variables for forecasting RV. Once more, this changing predictive power over time emphasises the importance of including a broad and diverse array of predictors in the ML models for RV forecasting.

The findings and conclusions remain the same after a series of robustness checks. The ML models continue to dominate all HAR-family of models for 90% of the out-of-sample forecasting period when the actual RV was not too extreme. Our robustness checks also revealed a couple of complex subtleties. The first subtlety concerns the ML specification. For the 90% out-of-sample forecasting period when actual RV is within the normal level, a simple ML structure with a small number of units and a small number of epochs prevails. For the 10% of the forecasting period when volatility is extremely high, a more complex ML structure produced a better forecasting performance. The second subtlety concerns QLIKE as the loss function. Despite the emphasis on preventing large under-forecast, QLIKE is a poor loss function as the weights become flat at high volatility, creating resistance for producing high forecast values; therefore, the optimisation algorithm loses the ability to learn to forecast accurately.

This paper provides a new understanding of the complexities behind developing ML models for volatility forecasting. The tuning of the ML hyperparameters, the choice of the loss function, and the information content of a large volume of input variables can all affect the forecasting performance conditioned on the level of actual RV. This study is the first comprehensive assess-

ment of employing a rich feature set for RV forecasting and lays the foundation for future ML research in this area. Future research could focus on extending the forecast evaluation to include multi-step ahead RV forecasts. Future research could also focus on forecasting high volatility days and implement some switching signals for modelling adjustments reflecting the ongoing volatility conditions. Furthermore, tree-based models like Xgboost (Chen and Guestrin, 2016) and LGBM (Ke et al., 2017) present a promising direction for future investigations. Tree-based models could potentially lead to improved training speed and a reduction in the number of parameters, while preserving the flexibility of nonlinearity.[5] Another research direction involves the development of a universal model for RV forecasting, potentially by leveraging all the available information in the market.

[5]We would like to thank an anonymous referee for this suggestion.

References

Andersen, T. G., T. Bollerslev, and F. X. Diebold (2007). Roughing it up: Including jump components in the measurement, modeling, and forecasting of return volatility. *The review of economics and statistics 89*(4), 701–720.

Andersen, T. G., T. Bollerslev, F. X. Diebold, and P. Labys (2001). The distribution of realized exchange rate volatility. *Journal of the American statistical association 96*(453), 42–55.

Andrychowicz, O. M., B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, et al. (2020). Learning dexterous in-hand manipulation. *The International Journal of Robotics Research 39*(1), 3–20.

Audrino, F. and S. D. Knaus (2016). Lassoing the har model: A model selection perspective on realized volatility dynamics. *Econometric Reviews 35*(8-10), 1485–1521.

Bali, T. G., H. Beckmeyer, M. Moerke, and F. Weigert (2021). Option return predictability with machine learning and big data. *Georgetown McDonough School of Business Research Paper* (3895984).

Barndorff-Nielsen, O. E., P. R. Hansen, A. Lunde, and N. Shephard (2009). Realized kernels in practice: Trades and quotes.

Barndorff-Nielsen, O. E. and N. Shephard (2001). Non-gaussian ornstein–uhlenbeck-based models and some of their uses in financial economics. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63*(2), 167–241.

Barndorff-Nielsen, O. E. and N. Shephard (2002). Estimating quadratic variation using realized variance. *Journal of Applied econometrics 17*(5), 457–477.

Barndorff-Nielsen, O. E. and N. Shephard (2004). Power and bipower variation with stochastic volatility and jumps. *Journal of financial econometrics 2*(1), 1–37.

Bollerslev, T., A. J. Patton, and R. Quaedvlieg (2016). Exploiting the errors: A simple approach for improved volatility forecasting. *Journal of Econometrics 192*(1), 1–18.

Bucci, A. (2020). Realized volatility forecasting with neural networks. *Journal of Financial Econometrics 18*(3), 502–531.

Chen, L., M. Pelger, and J. Zhu (2019). Deep learning in asset pricing. *Available at SSRN 3350138*.

Chen, T. and C. Guestrin (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794.

Christensen, K., M. Siggaard, and B. Veliyev (2022). A machine learning approach to volatility forecasting. *Journal of Financial Econometrics*.

Chronopoulos, I. C., A. Raftapostolos, and G. Kapetanios (2023). Forecasting value-at-risk using deep neural network quantile regression. *Journal of Financial Econometrics*.

Corsi, F. (2009). A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics 7*(2), 174–196.

Corsi, F. and R. Reno (2009). Har volatility modelling with heterogeneous leverage and jumps. *Available at SSRN 1316953*.

Diebold, F. and R. Mariano (1995). Comparing predictive accuracy. *Journal of Business Economic Statistics 13*(3), 253–63.

Erel, I., L. H. Stern, C. Tan, and M. S. Weisbach (2021). Selecting directors using machine learning. *The Review of Financial Studies 34*(7), 3226–3264.

Fernandes, M., M. C. Medeiros, and M. Scharth (2014). Modeling and predicting the cboe market volatility index. *Journal of Banking & Finance 40*, 1–10.

Goldberg, Y. (2016). A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research 57*, 345–420.

Gu, S., B. Kelly, and D. Xiu (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies 33*(5), 2223–2273.

Gu, S., B. Kelly, and D. Xiu (2021). Autoencoder asset pricing models. *Journal of Econometrics 222*(1), 429–450.

Hillebrand, E. and M. C. Medeiros (2010). The benefits of bagging for forecast models of realized volatility. *Econometric Reviews 29*(5-6), 571–593.

Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 6*(02), 107–116.

Hochreiter, S. and J. Schmidhuber (1997). Long short-term memory. *Neural computation 9*(8), 1735–1780.

Jiang, J., B. T. Kelly, and D. Xiu (2020). (re-) imag (in) ing price trends. *Chicago Booth Research Paper* (21-01).

Kalay, A., O. Sade, and A. Wohl (2004). Measuring stock illiquidity: An investigation of the demand and supply schedules at the tase. *Journal of Financial Economics 74*(3), 461–486.

Ke, G., Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems 30*.

Kercheval, A. N. and Y. Zhang (2015). Modelling high-frequency limit order book dynamics with support vector machines. *Quantitative Finance 15*(8), 1315–1329.

Kingma, D. P. and J. Ba (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Loshchilov, I. and F. Hutter (2017). Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Loughran, T. and B. McDonald (2011). When is a liability not a liability? textual analysis, dictionaries, and 10-ks. *The Journal of Finance 66*(1), 35–65.

Loughran, T. and B. McDonald (2016). Textual analysis in accounting and finance: A survey. *Journal of Accounting Research 54*(4), 1187–1230.

Loughran, T. and B. McDonald (2020). Textual analysis in finance. *Annual Review of Financial Economics 12*, 357–375.

Lundberg, S. M. and S.-I. Lee (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pp. 4768–4777.

Næs, R. and J. A. Skjeltorp (2006). Order book characteristics and the volume–volatility relation: Empirical evidence from a limit order market. *Journal of Financial Markets 9*(4), 408–432.

Patton, A. J. (2011). Volatility forecast comparison using imperfect volatility proxies. *Journal of Econometrics 160*(1), 246–256.

Patton, A. J. and K. Sheppard (2015). Good volatility, bad volatility: Signed jumps and the persistence of volatility. *Review of Economics and Statistics 97*(3), 683–697.

Politis, D. N. and J. P. Romano (1994). The stationary bootstrap. *Journal of the American Statistical association 89*(428), 1303–1313.

Poon, S.-H. and C. W. Granger (2003). Forecasting volatility in financial markets: A review. *Journal of economic literature 41*(2), 478–539.

Rahimikia, E. and S.-H. Poon (2022). Alternative data for realised volatility forecasting: Limit order book and news stories. *Available at SSRN 3684040*.

Shrikumar, A., P. Greenside, and A. Kundaje (2017). Learning important features through propagating activation differences. In *International Conference on Machine Learning*, pp. 3145–3153. PMLR.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research 15*(1), 1929–1958.

Vinyals, O., I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, et al. (2019). Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature 575*(7782), 350–354.

West, K. D. (1996). Asymptotic inference about predictive ability. *Econometrica: Journal of the Econometric Society*, 1067–1084.

White, H. (2000). A reality check for data snooping. *Econometrica 68*(5), 1097–1126.

A LOBSTER Database and Data Cleaning

The *LOBSTER* dataset is used for extracting the HAR-family variables in Subsection 3.1, and the Limit Order Book variables in Subsection 3.3. Before calculating these variables, for the preprocessing of the *LOBSTER* dataset, the proposed modified cleaning steps in Rahimikia and Poon (2022) based on Barndorff-Nielsen et al. (2009) are used in this study. These steps are applied to both the limit order book and message data in the following way (the step names, such as P2, T4, ..., are consistent with the names in Barndorff-Nielsen et al. (2009)):

- **P2:** Delete entries with a bid, ask or transaction price equal to zero.
- **T4:** Delete entries with prices that are above the 'ask' plus the bid-ask spread, or below the 'bid' minus the bid-ask spread.
- **Q1:** When multiple quotes share the same timestamp, they are replaced by a single entry using the median bid price, median ask price, and the sum of all volumes from these multiple quotes. For messages with the same direction (buy or sell), the mentioned procedure is applied to the message data, and the last snapshot of the LOB is selected as the LOB associated with the merged message data. For messages with different directions, the message data and the LOB with the same direction are grouped, and the mentioned procedure is applied separately to the buy-side and sell-side.
- **Q2**: Delete entries for which the spread is negative.
- **Q3**: Delete entries for which the spread is more than 50 times the median spread on that day.
- **Q4**: Delete entries for which the mid-quote deviated by more than 10 mean absolute deviations from a rolling centred median (excluding the observation under consideration) of 50 observations (25 observations before and 25 after).

The data cleaning summary statistics is summarised in Table A1.

Table A1: Data cleaning summary statistics
Sample period from 27 July 2007 to 27 January 2022

Name	Ticker	Sample size	Cleaned (%)	P2 (%)	T4 (%)	Q1 (%)	Q2 (%)	Q3 (%)	Q4 (%)
Apple	AAPL	4174971328	34.22	0.00	0.00	4.34	29.88	0.00	0.00
Microsoft	MSFT	3827824574	35.88	0.01	0.00	5.69	30.18	0.00	0.00
Intel	INTC	2807965330	38.59	0.01	0.01	6.78	31.79	0.00	0.01
Comcast	CMCSA	2390133817	45.18	0.01	0.00	5.58	39.59	0.00	0.01
Qualcomm	QCOM	2086295132	41.46	0.00	0.00	4.98	36.46	0.00	0.01
Cisco Systems	CSCO	2296179428	40.46	0.01	0.00	6.94	33.50	0.00	0.01
eBay	EBAY	1683001942	40.73	0.01	0.00	5.03	35.68	0.00	0.01
Gilead Sciences	GILD	1404574567	41.68	0.00	0.00	3.25	38.41	0.00	0.010
Texas Instruments	TXN	1485049597	39.45	0.00	0.00	4.29	35.14	0.00	0.01
Amazon.com	AMZN	1201210867	23.06	0.00	0.00	3.15	19.89	0.00	0.02
Starbucks	SBUX	1564221129	44.04	0.01	0.00	4.07	39.95	0.00	0.01
Nvidia	NVDA	1548447223	35.47	0.01	0.00	5.05	30.40	0.00	0.01
Micron Technology	MU	2110482619	35.99	0.00	0.00	4.86	31.11	0.00	0.01
Applied Materials	AMAT	1616466522	39.70	0.01	0.00	5.27	34.41	0.00	0.01
NetApp	NTAP	1015914054	44.99	0.01	0.00	3.82	41.14	0.00	0.02
Adobe	ADBE	1083392595	37.76	0.01	0.00	3.39	34.35	0.00	0.02
Xilinx	XLNX	1172584895	40.24	0.01	0.00	3.26	36.97	0.00	0.02
Amgen	AMGN	863464001	38.62	0.01	0.00	3.86	34.73	0.00	0.02
Vodafone Group	VOD	1012861232	47.20	0.01	0.00	2.95	44.23	0.00	0.02
Cognizant	CTSH	928987253	46.22	0.01	0.00	2.91	43.28	0.00	0.02
KLA Corporation	KLAC	783931409	42.63	0.01	0.00	2.77	39.83	0.00	0.02
Paccar	PCAR	775954122	45.74	0.01	0.00	2.73	42.98	0.00	0.03
Autodesk	ADSK	803552017	41.73	0.01	0.00	2.74	38.96	0.00	0.02
Average			40.05	0.01	0.00	4.25	35.78	0.00	0.01

Notes: **P2:** Delete entries with a bid, ask or transaction price equal to zero, **T4:** Delete entries with prices that are above the 'ask' plus the bid-ask spread, or below the 'bid' minus the bid-ask spread, **Q1:** When multiple quotes share the same timestamp, they are replaced by a single entry using the median bid price, median ask price, and the sum of all volumes from these multiple quotes. For messages with the same direction (buy or sell), the mentioned procedure is applied to the message data, and the last snapshot of the LOB is selected as the LOB associated with the merged message data. For messages with different directions, the message data and the LOB with the same direction are grouped, and the mentioned procedure is applied separately to the buy-side and sell-side, **Q2**: Delete entries for which the spread is negative, **Q3**: Delete entries for which the spread is more than 50 times the median spread on that day, **Q4**: Delete entries for which the mid-quote deviated by more than 10 mean absolute deviations from a rolling centred median (excluding the observation under consideration) of 50 observations (25 observations before and 25 after).

B Full Out-of-Sample Period Radar Plots

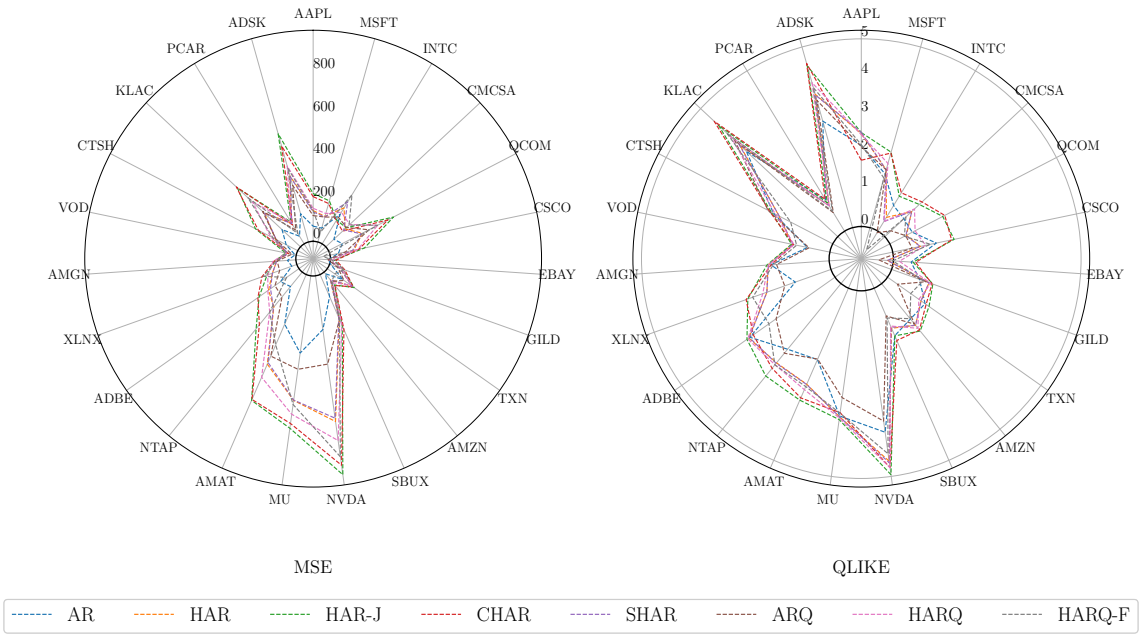


Figure B1: $\Delta_{MSE,i}$ and $\Delta_{QLIKE,i}$ between OB-ML (25 units) and HAR-family of models (full out-of-sample period)

Notes: Every radar chart contains the difference between the performance of the OB-ML model with 25 units with the AR1, HAR, HAR-J, CHAR, SHAR, ARQ, HARQ, and HARQ-F models for the mentioned tickers throughout the entire out-of-sample period. The left and right radar charts contain results for the MSE and QLIKE loss functions, respectively. For these loss functions, the negative value shows improvement. The bold circle inside these radar charts shows no improvement (zero).

C MDA Results for Primary Experiment

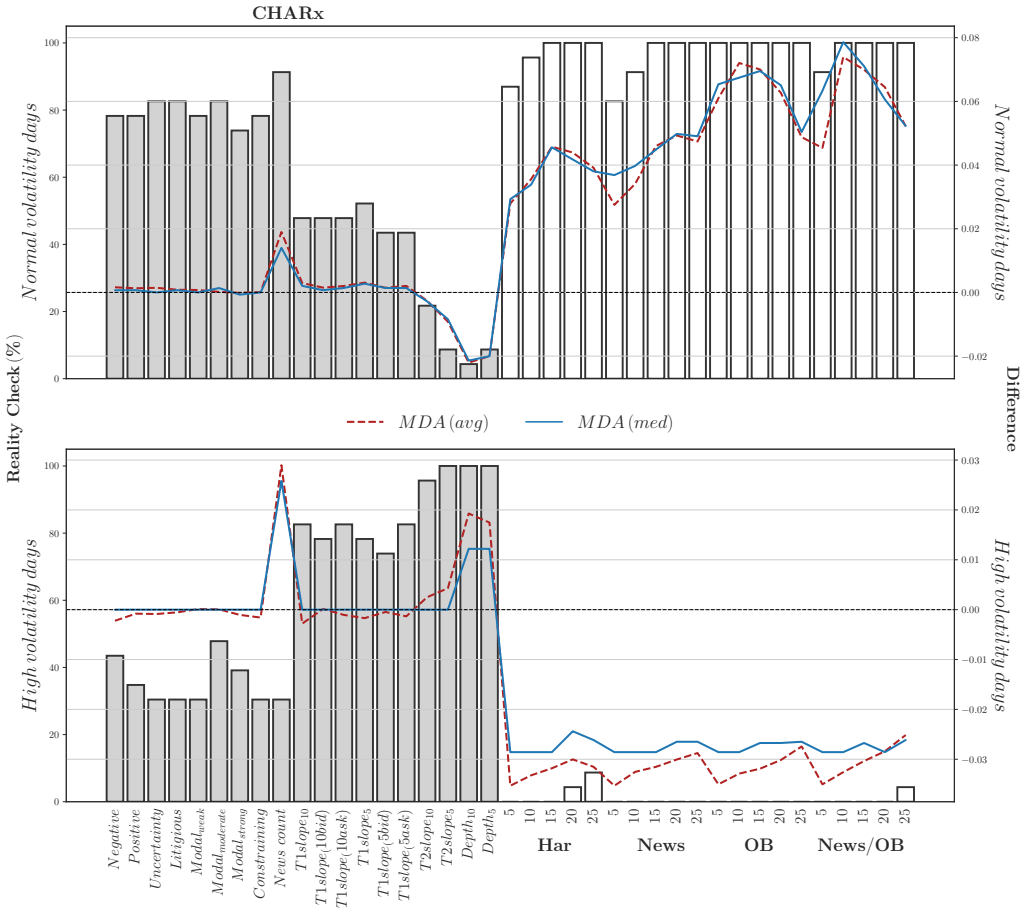


Figure C1: ML models and extended CHAR models comparison (MDA)

Notes: The bar chart is the percentage of tickers with outstanding performance considering the MDA loss function at the 5% significance level of the RC compared to the all HAR-family of models as the benchmark for every specified extended CHAR model (hatched bars) and the ML model (white bars). The values for the bar chart can be read from the left-hand axis. The dashed (solid) line represents the difference between the average (median) of the out-of-sample MDAs of the specified model with the CHAR model for 23 tickers (positive value shows improvement, and negative value shows degradation in performance). The values for the dashed and solid lines can be read from the right-hand axis. The horizontal dashed line represents no improvement.

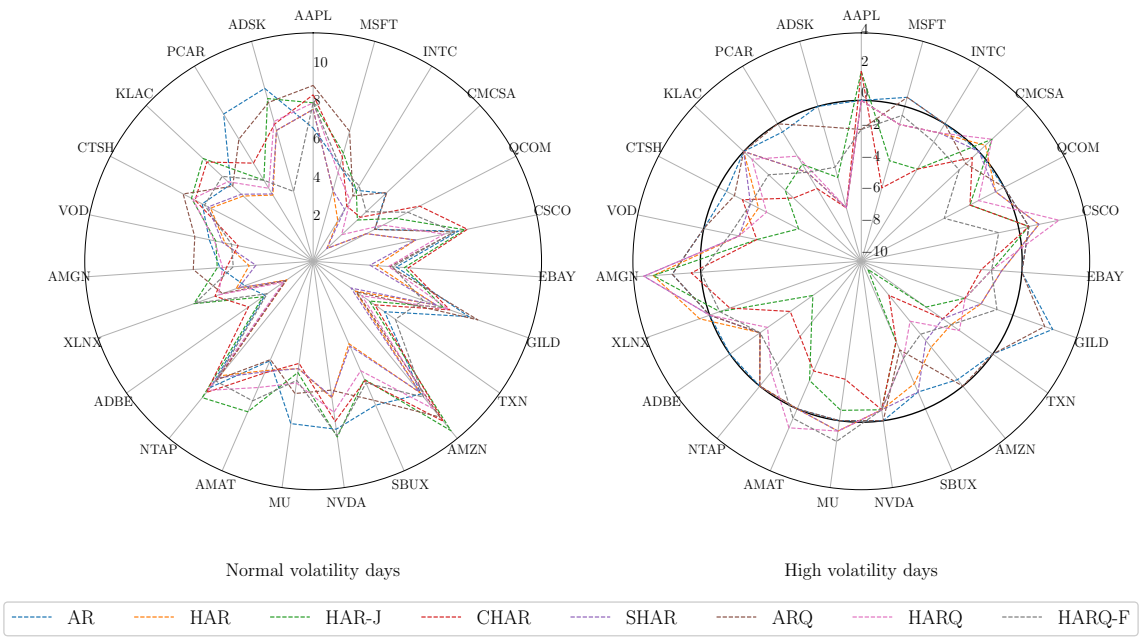
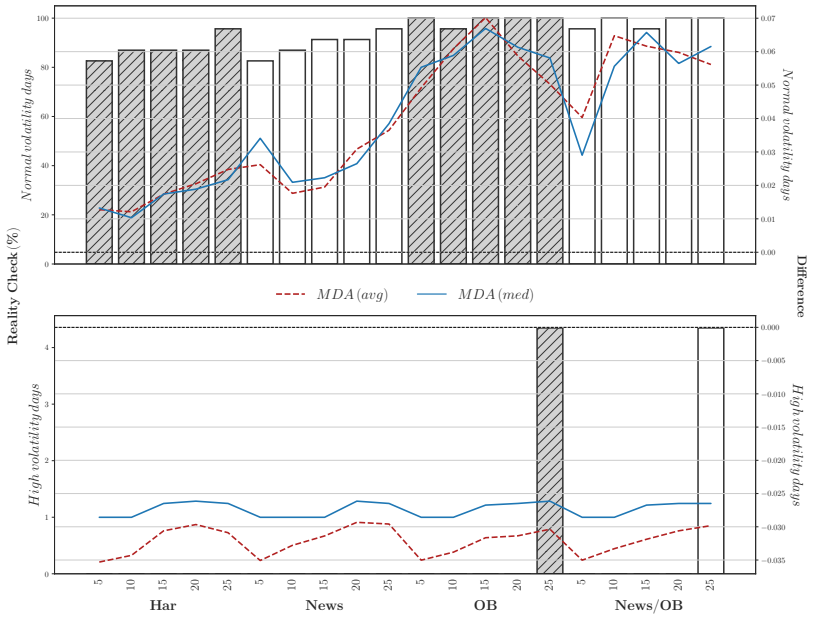


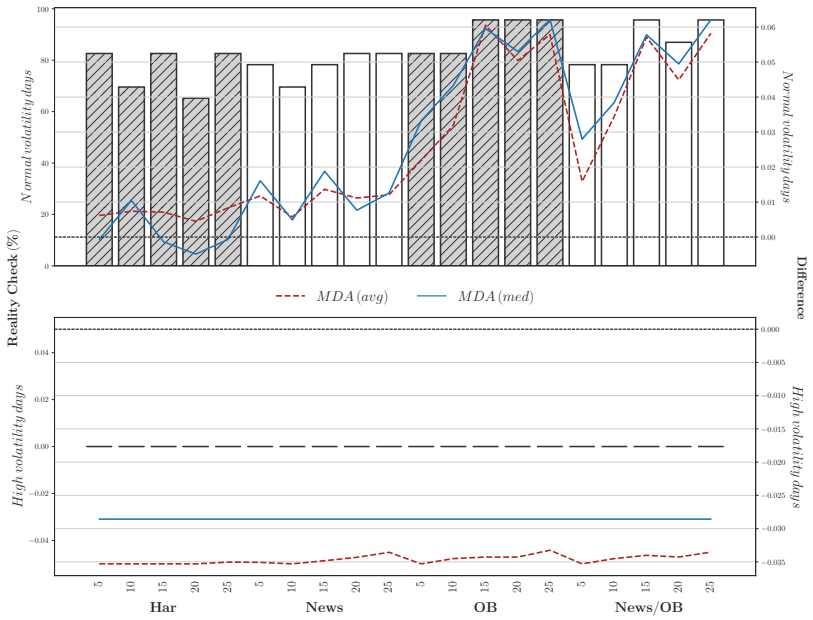
Figure C2: $\Delta_{MDA,i}$ between OB-ML (15 units) and HAR-family of models

Notes: Every radar chart contains the difference between the MDA performance of the OB-ML model with 15 units with the AR1, HAR, HAR-J, CHAR, SHAR, ARQ, HARQ, and HARQ-F models for the mentioned tickers. The left and right radar charts contain results of normal and high volatility days, respectively. For these radar charts, the positive value shows improvement, and the negative value shows degradation in performance. The bold circle inside of these radar charts shows no improvement (zero).

D MDA Results for Robustness Checks



(a) Five lags



(b) One lag

Figure D1: ML models with restricted information set $\Phi_t = \{t, ..., t-4\}$ (MDA)

Notes: The left and right figures display the results considering five lags (last week) and one lag (last day). The bar chart is the percentage of tickers with the outstanding performance considering the MDA loss function at the 5% significance level of the RC compared to the HAR-family of models as the benchmark for every HAR-ML, News-ML, OB-ML, and News/OB-ML model. The values for the bar chart can be read from the left-hand axis. The dashed (solid) line represents the difference between the average (median) of the out-of-sample MDAs of the HAR-ML, News-ML, OB-ML, and News/OB-ML models and the CHAR model for 23 tickers (positive value shows improvement, and negative value shows degradation in performance). The values for the dashed and solid lines can be read from the right-hand axis. The horizontal dashed line represents no improvement.

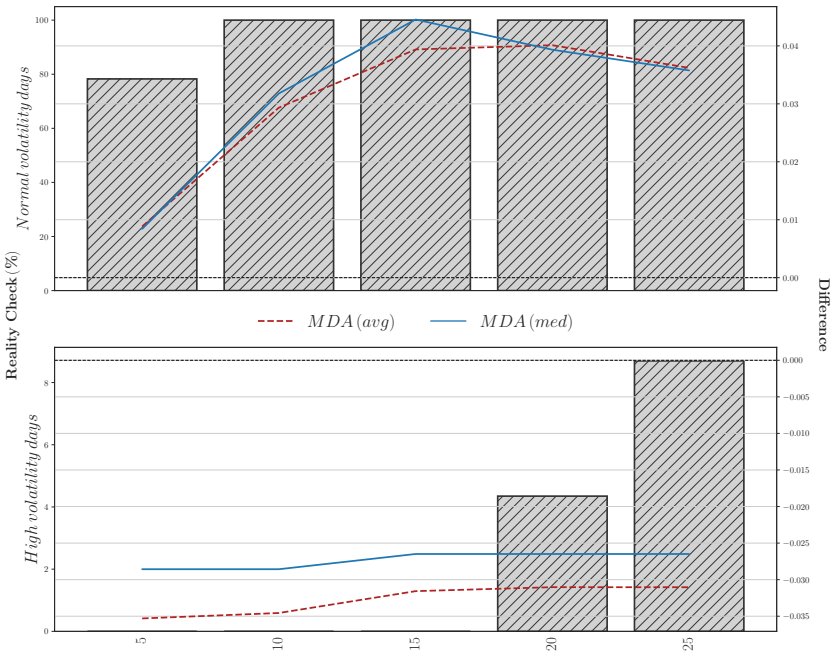


Figure D2: LSTM with $\Phi_t = \{RV_t, ..., RV_{t-20}\}$ (MDA)

Notes: The bar chart is the percentage of tickers with outstanding performance considering the MDA loss function at the 5% significance level of the RC compared to the HAR-family of models as the benchmark for every specified number of units of the ML model. The values for the bar chart can be read from the left-hand axis. The dashed (solid) line represents the difference between the average (median) of the out-of-sample MDAs of the specified ML models and the CHAR model for 23 tickers (positive value shows improvement, and negative value shows degradation in performance). The values for the solid and dashed lines can be read from the right-hand axis. The horizontal dashed line represents no improvement.

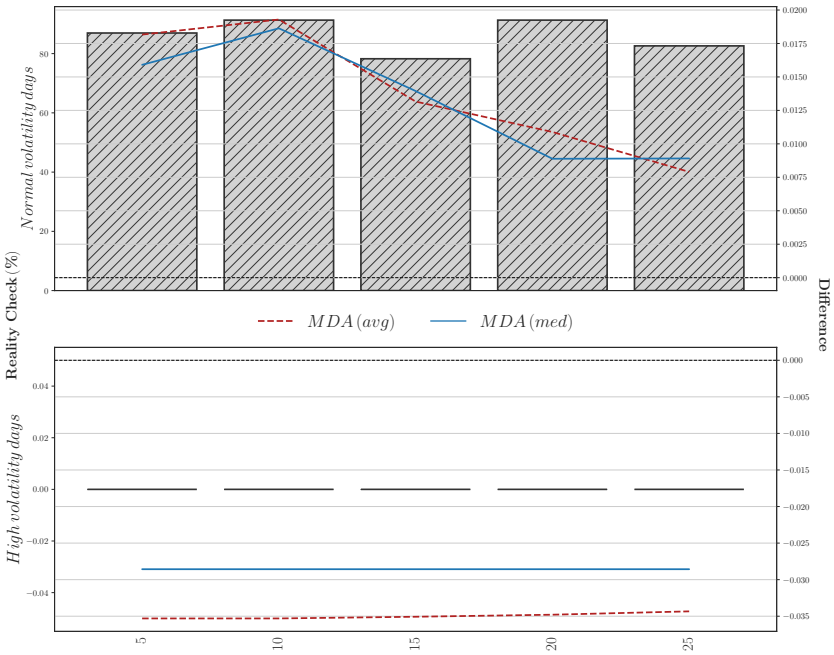
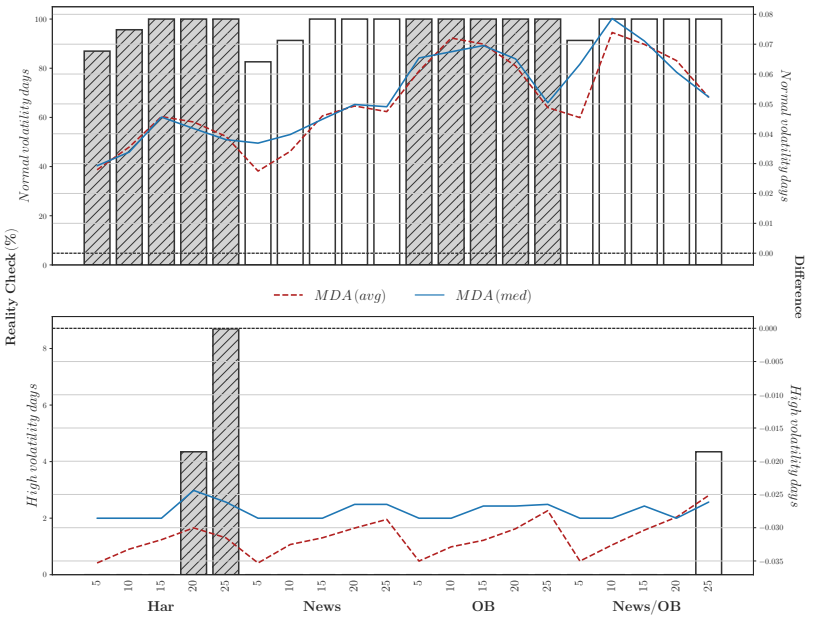
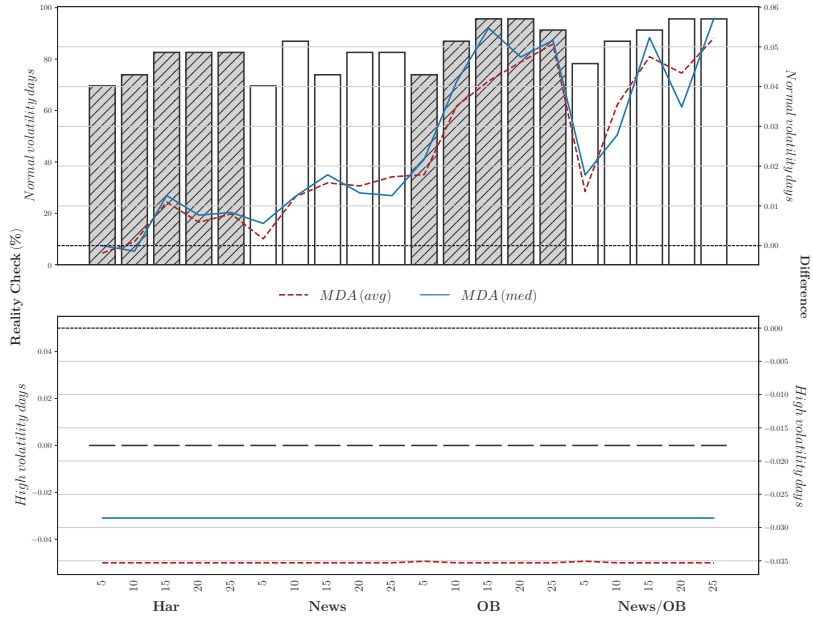


Figure D3: FCNN with $\Phi_t = \{RV_t, \overline{RV}^w_{t-1}, \overline{RV}^m_{t-1}\}$ (MDA)

Notes: The bar chart is the percentage of tickers with outstanding performance considering the MDA loss function at the 5% significance level of the RC compared to the HAR-family of models as the benchmark for every specified number of units of the ML model. The values for the bar chart can be read from the left-hand axis. The dashed (solid) line represents the difference between the average (median) of the out-of-sample MDAs of the specified ML models and the CHAR model for 23 tickers (positive value shows improvement, and negative value shows degradation in performance). The values for the solid and dashed lines can be read from the right-hand axis. The horizontal dashed line represents no improvement.



(a) MSE loss function



(b) QLIKE loss function

Figure D4: Minimising MSE vs QLIKE as the loss function in training period (MDA)

Notes: The bar chart is the RC percentage of tickers with outstanding ML performance at the 5% significance level against all the HAR-family of models. The values for the bar chart can be read from the left-hand axis. The red dashed (blue solid) line represents the average (median) Δ_{MDA} for each of the four ML models (HAR-ML, News-ML, OB-ML, and News/OB-ML) against CHAR for 23 tickers; a positive value indicates improvement, and a negative value indicates a performance degradation. The values for the dashed and solid lines can be read from the right-hand axis. The horizontal dashed line represents no difference in performance between ML and CHAR.

E Complementary Results

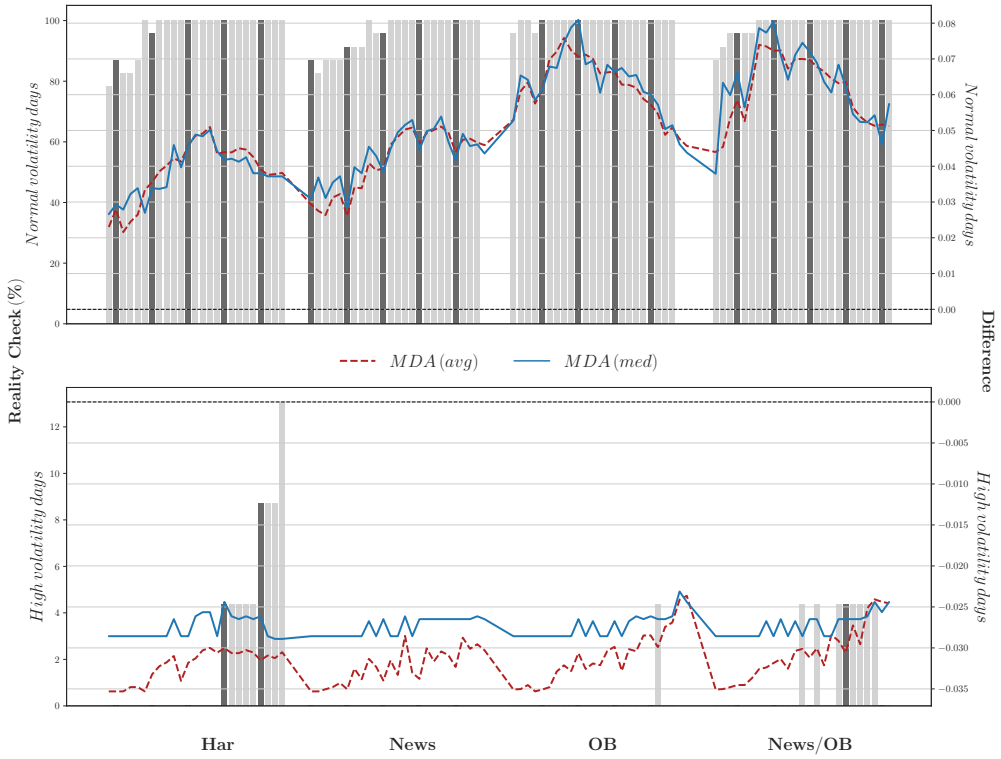


Figure E1: No of units vs No of epochs (MDA)

Notes: From left to right, this figure consists of the HAR-ML, News-ML, OB-ML, and News/OB-ML variable groups. For every group, the results are shown in the following order from left to right (#units-#epochs): 5-25, 5-50, 5-75, 5-100, 5-125, 10-25, 10-50, 10-75, 10-100, 10-125, 15-25, 15-50, 15-75, 15-100, 15-125, 20-25, 20-50, 20-75, 20-100, 20-125, 25-25, 5-50, 25-75, 25-100, and 25-125. For the sake of clarity, these values are not shown in this figure. The bar chart is the percentage of tickers with the outstanding performance considering the MDA loss function at the 5% significance level of the RC compared to the HAR-family of models as the benchmark for every specified ML model. The darker bar charts are the RC values of primary experiments in Section 5. The RC values can be read from the left-hand axis. The dashed (solid) line represents the difference between the average (median) of the out-of-sample MDAs of the specified ML model with the CHAR model for 23 tickers (positive value shows improvement, and negative value shows degradation in performance). The values for the dashed and solid lines can be read from the right-hand axis. The horizontal dashed line represents no improvement.