Comparing GARCH and NN for Forecasting TTF Volatility

Aalborg University Business School

Nicolaj Høgh Sørensen (20185522)

Cand Merc Finance, MSc, 4th Semester

Supervisor: Olesia Verchenko

June 1, 2023

Abstract

This thesis compares the predictive capabilities of econometric and machine learning models in forecasting volatility in gas spot data. The analysis incorporates varying interval windows (5-day, 10-day, and 22-day) to evaluate model performance and adaptability to data characteristics.

The research findings highlight the strengths and limitations of both model traditional econometric models and machine learning models. Traditional econometric models, such as Generalized Autoregressive Conditional Heteroskedasticity (GARCH) and GJR-GARCH, incrementally improved their performance with longer interval windows but consistently fell short of surpassing the prediction accuracy of other models. The GARCH models struggled to fully capture the complexities inherent in gas spot data volatility, indicating a need for more adaptive and nuanced modeling techniques. In contrast, machine learning models, particularly Long Short-Term Memory (LSTM) and bidirectional LSTM (BiLSTM) models emerged as the models' performing best in this comparative analysis. Across all interval windows, these models consistently demonstrated superior performance, often outperforming the GARCH models and the naive benchmark. Especially, the univariate LSTM model exhibited impressive consistency and robustness in its predictive accuracy. These findings suggest that machine learning models have an edge over traditional econometric models in forecasting volatility in gas spot data.

The thesis emphasizes the potential of machine learning techniques and highlights their adaptability to capture the intricate patterns within gas spot data volatility. Future studies should further explore and refine these machine-learning approaches to enhance volatility predictions in gas markets. By comparing econometric models to machine learning models, this thesis provides valuable insights into the prediction of volatility in gas spot data. The superiority of machine learning models underscores their potential as effective tools for accurate volatility forecasting, contributing to improved decision-making in gas markets.

Contents

1 Introduction

Volatility forecasting serves as an indispensable cornerstone in financial and commodity markets. It underpins various activities such as risk management, portfolio optimization, and derivative pricing. In the case of gas spot price volatility, predicting fluctuations becomes particularly crucial for market participants and policymakers. However, accurately capturing the complex nature of these price changes remains a challenging task. This necessitates the exploration of robust and accurate models.

This thesis offers a comprehensive assessment of various forecasting models used for predicting volatility in gas spot prices. By harnessing the potential of both traditional econometric models and advanced machine learning approaches, it aims to evaluate their relative performance and draw implications for future research. The thesis delves into the nuances of the Generalized Autoregressive Conditional Heteroskedasticity (GARCH) model and its asymmetric variant, GJR-GARCH, both widely applied in volatility forecasting. These models adeptly capture volatility clustering and utilize historical price information to forecast future volatility. Additionally, to these traditional models, a 'naive' forecasting model is included in the comparison. This simplistic yet often effective model, hinging on the belief that future patterns will mirror past behavior, serves as a baseline against which the performance of the other models. In contrast, the introduction of machine learning becomes quite important, specifically, the focus is on the Long Short-Term Memory (LSTM) and bidirectional LSTM (BiLSTM) models. These neural networks leverage their learning capabilities to model non-linear dependencies and patterns within the data, capturing the complexity of gas spot prices.

The analysis examines these models over different forecasting horizons and interval windows, providing a detailed performance comparison. The assessment is based on Root Mean Square Prediction Error (RMSPE) and Root Mean Square Error (RMSE), offering both relative and absolute measures of forecasting accuracy, as well as a Diebold-Mariano (DM) test for significant differences of the forecasts. Throughout this analysis of the forecast accuracy of the 'naive', GARCH, GJR-GARCH, LSTM, and BiLSTM models, interesting discoveries are made that suggest reconsidering how gas spot price volatility is modeled. The thesis concludes with a reflection on the findings and proposes future directions that hold the potential for further refining forecasting accuracy.

1.1 Introduction to Natural Gas and Importance of Topic

Natural gas is an essential energy commodity and plays a significant role in the European energy market (IEA, 2019). It is the cleanest-burning fossil fuel and its usage for heating, electricity generation, and diverse industrial applications has been on the rise, thanks to its lower carbon emissions compared to coal and oil (EIA, 2021). This has influenced demand dynamics and thus the price volatility in spot markets, which are primarily dictated by immediate supply-demand conditions. Despite being the same commodity, the spot and derivatives markets for natural gas operate distinctly due to the logistics involved in production, transportation, and storage. This separation also contributes to the price-setting mechanisms in these markets, with spot prices exhibiting more volatility due to limited time to adjust supply or demand. Moreover, the natural gas market is influenced by broader geopolitical scenarios, environmental policies, and technological advancements (IEA, 2019). Countries with significant natural gas reserves or control over strategic supply routes often wield considerable geopolitical power. Additionally, the global transition towards lower-carbon energy sources and the development of advanced extraction technologies like hydraulic fracturing influence the demand and supply of natural gas, respectively (EIA, 2021).

In Europe, natural gas is traded at various hubs and exchanges, with the Title Transfer Facility (TTF) in the Netherlands and the National Balancing Point (NBP) in the United Kingdom standing out as key marketplaces (ENTSOG, 2020). Alongside these hubs, the Henry Hub, located on the U.S. Gulf Coast, sets the benchmark for all natural gas traded in the United States (EIA, 2021). The dynamics of these hubs play a crucial role in setting price trends in both spot and forward markets.

The volatility of gas spot prices presents a complex challenge for traders, analysts, and policymakers, making the forecasting of these price movements a significant area of interest. The inherent complexity and high stakes of the energy markets have led to the development and application of advanced forecasting models, ranging from traditional econometric models like GARCH to cutting-edge machine learning models such as LSTM and BiLSTM (Chong et al., 2017). Although traditional models like GARCH models have proven effective, they may not fully capture the unique characteristics of the natural gas market, such as their distinctive seasonality and inverse leverage effect (Geman,

2005). These dynamics indicate the potential for models that can adapt to non-linear relationships and complex patterns. Machine learning models like LSTM and BiLSTM (Hochreiter & Schmidhuber, 1997).

While comparative studies between GARCH and ML models exist, comprehensive research specifically focused on natural gas volatility forecasting remains scarce. This thesis addresses this gap, contrasting the predictive accuracy of LSTM, BiLSTM, and GARCH models. The intention is not to discredit the longstanding GARCH models, but to explore and quantify the potential benefits offered by ML models in forecasting natural gas price volatility. Considering the constantly evolving nature of these markets and the increasing availability of granular data, this thesis investigates the performance of various forecasting models in predicting gas spot price volatility. The main objective is to provide valuable insights for traders, researchers, and policymakers alike, thus contributing to a deeper understanding of the dynamics governing natural gas markets.

1.2 Problem Statement

As stated above, natural gas is a crucial energy commodity that is traded in spot markets, where prices are influenced by a wide range of factors, including supply and demand, weather patterns, geopolitical events, global economic conditions, etc. The volatility in gas spot prices can create significant risks and opportunities for traders and investors, and accurate forecasting of volatility is essential for effective risk management and trading strategies. However, volatility in gas spot prices is notoriously difficult to predict, and existing models often suffer from various limitations and shortcomings.

Therefore, this thesis aims to develop a comprehensive framework for volatility modeling in TTF gas spot data comparing statistical and machine learning techniques. Specifically, this thesis will explore the following research questions:

- *How do statistical and machine learning models compare in terms of their accuracy and performance in predicting volatility in gas spot data, and how do their performance compare when analyzing varying interval windows?*

To answer these research questions, the thesis will utilize historical gas spot data and apply various econometric and machine learning techniques, such as GARCH models and neural networks. The performance of these models will be evaluated based on relative and

absolute performance measurements, as well as through a DM test. As will be assessed later, this thesis will utilize univariate models to investigate the effect of less complex models and to explore the value of simplicity in forecasting gas spot volatility.

This thesis aims to contribute to the field of finance by providing a more accurate and comprehensive framework for volatility modeling in gas spot data through machine learning models, and how the performance of these models compare to traditional econometric models.

2 Philosophy of Science

This chapter explores the philosophical underpinnings of the research on volatility forecasting using neural networks and econometric models from a critical realist perspective. The following explanations of paradigms, ontology, and epistemology are based on Egholm (2014). Critical realism is a post-positivist paradigm that acknowledges the complexity of social phenomena while emphasizing the importance of empirical observation and the existence of objective patterns. This chapter will discuss the epistemological, ontological, methodological, and ethical dimensions of the research within the critical realist framework.

2.1 Epistemology

Critical realism asserts that knowledge can be obtained through the empirical observation of objective patterns in the world. In the context of volatility forecasting, this perspective supports the notion that financial markets exhibit complex and nonlinear patterns that can be uncovered through the analysis of historical financial data. This thesis relies on the use of neural networks and GARCH models to predict and model these patterns, emphasizing the importance of empirical observation and data-driven approaches in generating knowledge about volatility forecasting.

2.2 Ontology

Critical realism posits that reality consists of multiple layers, with complex causal mechanisms and structures that may not be directly observable. This ontological perspective is well-suited for studying financial markets, which are characterized by intricate relationships between economic factors, investor behavior, and market dynamics. In this thesis, neural networks are used to model the complex, nonlinear relationships that underlie volatility patterns in financial markets, acknowledging the layered and multifaceted nature of reality.

2.3 Ethical Considerations and Implications of Philosophy of Science

Critical realism recognizes the potential ethical implications and consequences of research findings. In the context of volatility forecasting using neural networks, ethical considerations include the importance of transparency and accountability in financial modeling. This thesis, grounded in the critical realist perspective, highlights the potential of neural networks as powerful tools for modeling complex patterns in financial markets, while also emphasizing the need for continuous methodological advancements and interdisciplinary collaboration. The critical realist paradigm provides a strong philosophical foundation for this thesis, acknowledging the complexity of financial markets and the importance of empirical observation, quantitative methods, and the pursuit of objective patterns and causal mechanisms in volatility forecasting.

By adopting a critical realist perspective, this Philosophy of Science chapter provides a deeper understanding of the assumptions, methods, and implications underlying the research on volatility forecasting using neural networks. This philosophical framework recognizes the complexity of financial markets and supports the use of advanced computational techniques, such as neural networks, for uncovering objective patterns and improving forecasting accuracy. By incorporating critical realist principles, this study contributes to the ongoing development and refinement of volatility forecasting methods, while also addressing ethical considerations and promoting responsible use in the finance industry.

3 Literature Review

In this literature review, the significance of volatility forecasting in finance and the growing application of neural networks in this field will be explored. As traditional models have shown limitations, neural networks have emerged as a promising alternative, offering better predictive accuracy and adaptability to nonlinear patterns in financial data. Thus, this literature review will investigate the research of various authors on volatility forecasting and neural networks in general and within the field of finance. Ultimately, this section will focus on any gaps and future research that is present in this fast-advancing field.

3.1 Neural Networks

Neural networks are inspired by the human brain and consist of interconnected nodes or neurons. They have been widely adopted in finance due to their ability to learn and model complex patterns. Early applications include stock price prediction (Kimoto et al., 1990) and bankruptcy prediction (Altman et al., 1994). The development of deep learning techniques (Bengio et al., 2015) has further enhanced their capabilities and applications in finance. Kimoto et. al (1990), being one of the earliest academic applications of neural networks, studied the stock market, and developed a learning and prediction algorithm for the Tokyo Stock Exchange Prices Indexes (TOPIX). The model consisted of multiple inputs, such as foreign exchange rate, interest rate, turnover rate, the New York Dow Jones average, etc. The researchers' prediction algorithm gained an edge on the TOPIX by 30.1 percentage points over a 2.5-year period, as the simple Buy-and-Hold strategy yielded a 67% return, whereas the prediction model yielded a 98% return. The study created a base for future research on building more accurate economic prediction algorithms.

Furthermore, within the field of finance, various researchers have applied various types of neural networks for volatility forecasting, including Forward Feeding Neural Networks (FFNN) (Kaastra & Boyd, 1996), Recurrent Neural Networks (RNN) (Zhang & Zhang, 2008), LSTM (Fischer & Krauss, 2018), and Gated Recurrent Unit (GRU) (Cho et al., 2014). Fischer and Krauss (2018) investigated the directional movements using Long short-term memory (LSTM) network, of the constituents of the S&P 500 from 1992 until 2015. Thus this paper explores the application of LSTM networks for financial market

predictions, including volatility forecasting. This study used only time series data, contrary to eg. Kimoto et. al (1990), who used mainly economic input. The study found that the LSTM prediction model significantly outperforms other prediction models including GARCH models. The researchers were able to construct a short-term reversal trading strategy that yielded a 0.23 percent daily return prior to transaction costs. The result demonstrates that RNN can provide a more accurate volatility forecast compared to traditional linear models. These studies have thus generally implied that various neural network models outperform traditional models in terms of predictive accuracy. However, they also highlight the need for large amounts of data and computational power, and the challenge of interpreting the models' inner workings.

3.2 Volatility

Volatility represents the degree of variation in asset prices over time, reflecting market uncertainty and risk. Thus, volatility does not measure the direction of price changes of a given financial instrument, but rather the dispersion of the price over a certain period of time.

Accurate volatility forecasting is essential for risk management, option pricing, and portfolio optimization. Traditional methods of volatility forecasting include autoregressive models like GARCH (Bollerslev, 1986) and stochastic volatility models (Hull and White, 1987). However, these models often struggle to capture complex, nonlinear patterns in financial data. In the context of financial markets, volatility is typically categorized into two distinct types: implied volatility and realized volatility.

Implied volatility and realized volatility are two important concepts in the field of financial markets, particularly in the context of options pricing and risk management. Understanding the differences between the two can help investors and traders make informed decisions about their strategies. Even though this thesis only revolves around realized volatility, the definition of implied volatility and its usage will greatly improve the overall understanding of volatility. Implied volatility (IV) is a measure of the expected future volatility of an underlying asset, derived from the prices of options on that asset. It represents the market's expectation of how much the asset's price will fluctuate over a specified period. Implied volatility is a crucial input for option pricing models like

the Black-Scholes model (Black & Scholes, 1973). It is often used as a gauge of market sentiment, with higher implied volatility indicating increased uncertainty and potential risk. Realized volatility (RV), also known as historical volatility, refers to the actual fluctuations in an asset's price over a specified period in the past. It is calculated by measuring the standard deviation of the asset's price returns over a given time frame. Realized volatility serves as a useful benchmark for evaluating the accuracy of volatility forecasts, as it represents the true historical volatility experienced by market participants (Andersen et. al (2003).

Both implied and realized volatility play significant roles in financial markets, with implied volatility informing market participants about the expected future price fluctuations, while realized volatility provides a historical record of actual price movements. Accurate forecasts of volatility, whether implied or realized, are essential for risk management, option pricing, and portfolio optimization. This makes the development of advanced forecasting models, such as those based on neural networks, a critical area of research for improving the understanding of market dynamics and enhancing investment strategies. Even though implied volatility is an essential part of option pricing, and thus an important aspect of the definition of volatility as a whole, this thesis does not work with implied volatility, whereby this will not be mentioned further, and whenever volatility is mentioned from this stage on, it is understood as realized or historical volatility.

3.3 Volatility Forecast

Volatility trading involves taking positions based on the expected level of market volatility, rather than the direction of asset prices. Accurate volatility forecasts can provide valuable insights for trading strategies, allowing traders to adjust their risk exposure and optimize returns. Several studies have investigated various aspects of volatility trading, such as the use of different forecasting models, trading strategies, and their performance in different market conditions (Jacobs & Li, 2021; Dunis & Miao, 2006; Gao, 2017; Zahid et al., 2022). Jacobs and Li (2021) analyzed the use of machine learning models, including neural networks, in predicting option-implied volatility and found that they outperformed traditional models like GARCH. The study highlights the potential benefits of incorporating machine learning models into volatility trading strategies. Dunis and Miao (2006) examined the profitability of trading strategies based on GARCH

volatility forecasts in foreign exchange markets. The study revealed that GARCH-based trading strategies can yield profitable results and contribute to improved risk-adjusted returns. Gao (2017) focused on the performance of volatility trading strategies using realized volatility and implied volatility forecasts. The author found that strategies using a combination of realized and implied volatility forecasts generated better risk-adjusted returns compared to strategies relying solely on either realized or implied volatility forecasts. Zahid et al. (2022) investigated hybrid GARCH models in forecasting Bitcoin volatility showing that hybrid models could in fact produce an accurate forecast of Bitcoin's price volatility. Thus these studies imply the importance of accurate volatility forecasting in developing effective trading strategies and the potential advantages of using advanced models, such as neural networks, for this purpose, but also the possibilities for GARCH models to yield great volatility predictions.

3.3.1 Volatility Forecast in Natural Gas Markets

As established in section 1, volatility forecasting plays a significant role in natural gas markets due to the commodity's inherent price fluctuations, which are influenced by various factors, including supply and demand dynamics, weather conditions, geopolitical events, and storage levels. Accurate volatility forecasts can help market participants, such as producers, consumers, and traders, make informed decisions and manage risks associated with natural gas price fluctuations. (Edwards, 2017)

Few studies have investigated volatility trading in natural gas markets, especially with a focus on comparing different forecasting models and their impact on trading strategies. Čeperić et al. (2017) investigated short-term forecasting of the Henry Hub natural gas spot prices based on various traditional time series models compared to neural networks. They introduced a procedure for including multiple inputs, utilizing feature selection for model input selection. The result of the study showed that the successfulness of neural networks models, proposed by the literature, was often exaggerated, and only recorded a slight improvement over traditional time series models. However, the study showed a significant advantage of the use of feature selection for the automatic selection of preselected variables for neural networks. Fałdziski et al. (2020) also compared the forecasting performance of GARCH-type models with machine learning models, in this case, support vector regression (SVR) for futures contracts of various selected commodities, including the Henry Hub

natural gas. Using squared daily returns as the proxy for volatility estimation, the SVR model showed lower forecasting error compared to the GARCH models. Moreover, the study had difficulty choosing a GARCH model that performed well across all commodities, however, forecasts based on the asymmetric variant of GARCH were often the most accurate.

This literature review implies the growing importance of neural networks in volatility forecasting, outperforming traditional methods in many cases. However, it also shows the capabilities traditional GARCH models have on forecasting volatility. Future research should focus on improving model architectures, enhancing the interpretation, and exploring real-world applications. By doing so, neural networks can continue to advance the field of volatility forecasting and contribute to more efficient financial markets. This thesis is inspired by Čeperić et al. (2017) comparing GARCH models with machine learning principles to forecast volatility in gas spot prices. Throughout the literature, there has been a tendency to include multiple variables to create complex multivariate models or probabilistic models when forecasting gas volatility. Most studies about forecasting natural gas aspect, either volatility or price, investigates Henry Hub natural gas data, whereby very little emphasis has been put on the European gas market. Furthermore, very little research has been done on forecasting volatility, especially natural gas volatility, using a univariate model. Therefore, this thesis will fill the gap in the research by investigating univariate forecasting models on the TTF gas spot prices. The focus on univariate models provides an opportunity to explore the value of simplicity and understand how much information is truly lost, or gained when additional variables are removed from the model. As such, this study will seek to evaluate the performance of these less complex models in capturing the dynamics of natural gas volatility in the European context, with the aim to offer valuable insights for future model selection and improvement. This could potentially simplify the forecasting process, making it more accessible to a broader range of stakeholders, and still maintaining an acceptable level of forecasting accuracy.

4 Model Specifications

In this chapter, the model specification of the five models that are used in the attempt to forecast volatility will be discussed. These models have been selected to represent a range of methodologies, from simplistic statistical models to more complex machine learning approaches. In the following sections, the specifications of these models are further assessed, starting with the baseline Naive model and progressing through to the more complex GARCH, GJR-GARCH, LSTM, and BiLSTM models.

4.1 Naive Random Walk Model

Given the autocorrelation and clustering historically observed in volatility data, a simplistic approach to forecasting is to use a naive random walk model. This model leverages the property of volatility persistence, which suggests that the immediate past volatility can be indicative of future volatility.

The naive random walk model can be mathematically expressed as follows:

$$\sigma_{t+i}^2 = \sigma_t^2 \epsilon_t$$

Where, σ_{t+1}^2 is the forecasted volatility for the next period (t+i), and σ_t^2 is the observed volatility at time t. The naive model works on the assumption that the future volatility will be the same as the immediate prior day's volatility. It relies entirely on this single feature and involves no advanced statistical or machine learning methodology, hence the term "naive".

Although simplistic in its construction, the naive random walk model serves a key purpose in our analysis. It provides a baseline for comparison, against which we can evaluate the performance of more sophisticated models like GARCH and LSTM. If our advanced models cannot outperform this naive model, it might suggest that these models are not well-tuned or that volatility in the natural gas market is exceptionally difficult to predict.

In time series analysis, the naive model often plays the role of a benchmark. Despite its simplicity, it can sometimes yield surprisingly good results, especially in cases where the data exhibits strong autocorrelation. (Tsay, 2010; Chatfield, 2005)

4.2 Generalized Autoregressive Conditional Heteroskedasticity (GARCH)

GARCH (Generalized Autoregressive Conditional Heteroskedasticity) is a statistical model used to estimate and forecast volatility in financial time series data. It was first introduced by Bollerslev (1986) as an extension to the ARCH (Autoregressive Conditional Heteroskedasticity) model, as it requires a substantial amount of lags to adequately describe the volatility process of an asset's return. The general formula denoted as GARCH(p,q) is stated as follows:

$$\sigma_t^2 = w + \sum_{i=1}^{q} \alpha_i r_{t-i}^2 + \sum_{i=1}^{p} \beta_j \sigma_{t-i}^2$$

where w is the variance at time step t, ϵ_{t-i}^2 is the model residuals at time step $t-i$, and α and β are parameters to be estimated. Thus, in this model, the conditional variance is defined as a function of the historical values of both the squared residuals and the conditional variance. Volatility persistence and volatility clustering are both captured by the GARCH model, however, time-dependent asymmetry is frequently left out in the standard GARCH models. Because of this, various extensions to the standard GARCH models have been proposed to accommodate asymmetric volatility characteristics.

4.2.1 Glosten-Jannathan-Runkle GARCH (GJR-GARCH)

The Glosten-Jannathan-Runkle GARCH (GJR-GARCH) is a variation of the GARCH proposed by Glosten et al. (1993). GJR-GARCH takes these asymmetric shocks into account, contrary to the standard GARCH which assumes that positive and negative volatility news have similar impact on volatility. The GJR-GARCH(1,1) models can be written as:

$$\sigma_t^2 = \omega + \alpha \cdot \epsilon_{t-1}^2 + \gamma \cdot \epsilon_{t-1}^2 \cdot I_{\epsilon_{t-1}<0} + \beta \cdot \sigma_{t-1}^2$$

in which ω, α, γ, and β are parameters to be estimated, ϵ_{t-1}^2 is the squared model residual at time step $t-1$, $I_{\epsilon_{t-1}<0}$ is an indicator function that equals 1 if $\epsilon_{t-1} < 0$, i.e., if the return at time $t-1$ was below the mean, and 0 otherwise, σ_t^2 is the conditional variance at time step t, σ_{t-1}^2 is the conditional variance at time step $t-1$.

In the field of finance, asymmetric shocks refer to the idea that "bad news" and "good news" can have different effects on a financial variable's volatility. For instance, a negative shock to a company's stock price (bad news) might increase the stock's volatility more than a positive shock of the same magnitude (good news). This is an empirical fact that is often observed in financial markets. Thus, the GJR-GARCH model improves upon the standard GARCH model by introducing an additional term that accounts for these asymmetric shocks. If the residual shocks are skewed either positively or negatively, the GJR-GARCH model can be a better choice for modeling and forecasting volatility. In the context of the thesis, after conducting the standard GARCH model, the distribution of the residuals will be evaluated. If the residuals are skewed, the GJR-GARCH model will be used for forecasting, as it can handle asymmetric shocks. By comparing the standard GARCH and GJR-GARCH models, it will be possible to assess the presence and impact of asymmetric shocks on the volatility of TTF Gas spot prices. This will also indicate if any asymmetry is present and if it will have any effect on the volatility in this case.

For this thesis, both GARCH models will implement a rolling forecast meaning the model will refit on a rolling basis, encompassing all available data points up to a specific time step. By adhering to this rolling one-step forecasting approach, the model continually incorporates the most up-to-date information in the data series, maximizing the precision of its short-term predictive output. However, it is acknowledged that this approach might lead to overfitting of the data, and risking the models might learn noise or specific details in the data that do not contribute to the general trend. However, one way to manage this risk would be to employ cross-validation, as dividing the data into training and validation sets enables monitoring the models' performance on unseen data and tuning the complexity of the models accordingly. Therefore, even though refitting the models at every time step might introduce the risk of overfitting, the benefit of quickly adapting to new data often outweighs the risk if these measures are taken into consideration. On another note, rolling forecasting is a commonly applied method in time series forecasting, as it allows for dynamic modeling (Bergmeir et. al, 2018; Hyndman & Athanasopoulos (2021).

4.3 Neural Networks

The sequencing models for predicting the TTF returns are built on the concept of Recurrent Neural Networks (RNN). RNN is a specific type of neural network consisting of multiple layers, known as recurrent layers. These layers operate sequentially and map the sequences to other sequences. Thus RNN retains information in its internal state, referred to as a memory cell. The output of RNN at a specific time interval depends on the input at that time interval and the network's state in the preceding time interval. However, one of the disadvantages of RNN is that it either stops learning or keeps learning at a high learning rate, resulting in the model being unable to grasp the concept of even the smallest error (Moghar & Hamiche, 2010). Therefore, standard RNN performance is not considered adequate when the learning requires long-term sequential dependencies and is thus considered for forecasting samples with long-time data (Hochreiter & Scmiduber, 1997). Thus better and more advance neural network models should be considered.

4.3.1 Long Short-Term Memory (LSTM)

The Long Short-Term Memory (LSTM) is a type of RNN that consists of a memory cell and various gates. The LSTM often provides better predictions than the RNN for longer sequential data, the structure of the LSTM networks makes it possible to forget past irrelevant information, thus this kind of network is more capable of modeling complex time series data. The figure below displays the architecture of the LSTM cell, including the special gates: forget gate, input gate, and output gate.

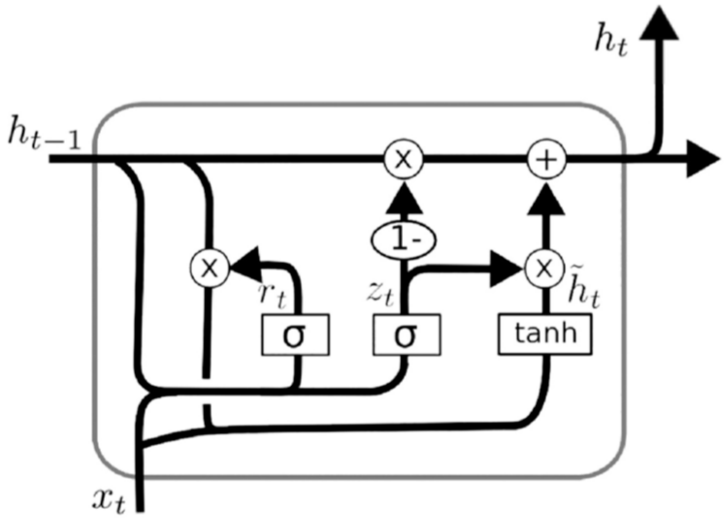

Figure 1: The Architecture of the LSTM Cell

The output gate controls what information should be output from the cell state to the next hidden state. This gate takes in the previous hidden state h_{t1} and the current input X_t combines them and passes through a sigmoid activation function. This generates a value σ_t between 0 and 1, which can be interpreted as the proportion of the cell state information that should be output to the next hidden state. The equation for the output gate is:

$$o_t = \sigma(W_o[h_{t-1}, X_t] + b_o)$$

In this equation, W_o is the weight matrix and b_o is the bias for the output gate, $ht1$ is the hidden state at time t-1, X_t is the input at time t, and σ represents the sigmoid function.

After the output gate, the cell state C_t is passed through a $tanh$ activation function to scale it between -1 and 1, and then it's multiplied by the output gate's output o_t to produce the final output ht of the LSTM cell. This output h_t is used as the hidden state for the next time step and is also the output of this LSTM cell for the current time step.

The equation for the final output value of the cell is:

$$h_t = o_t * \tanh(C_t)$$

In this equation, o_t is the output from the output gate, tanh is the hyperbolic tangent function (which scales the cell state to a value between -1 and 1), and Ct is the cell state at time t. The multiplication operation between o_t and $tanh(C_t)$ is element-wise if they are vectors or matrices. The output ht can be considered as the predicted value computed by the LSTM model for the current state, based on the information it has processed so far from the input sequence.

4.3.2 Bidirectional Long Short-Term Memory (BiLSTM)

The Bidirectional Long Short-Term Memory (BiLSTM), introduced by Shuster and Paliwal (1997), is a bidirectional variant of RNN that combines a forward and backward univariate LSTM as explained in the previous subsection. It is thus an extension of the traditional LSTM network. While univariate LSTM only can store long-term information from prior observations, BiLSTM uses the combined two hidden layers and is designed to improve model performance by learning from past (backward) and future (forward) data

in a sequence. In the BiLSTM architecture, for each moment t, the model has two hidden states: a forward state and a backward state. The forward state is calculated based on past information (up to time t), while the backward state is calculated based on future information (from time t onwards). Mathematically, this is represented as:

$$Concatenate(h_t) = [\overrightarrow{h_t}, \overleftarrow{h_t}]$$

Here, h_t represents the combined hidden state at time t, $\overrightarrow{h_t}$ is the forward hidden state, and $\overleftarrow{h_t}$ is the backward hidden state. The *Concatenate* operation simply combines these two states into a single vector.

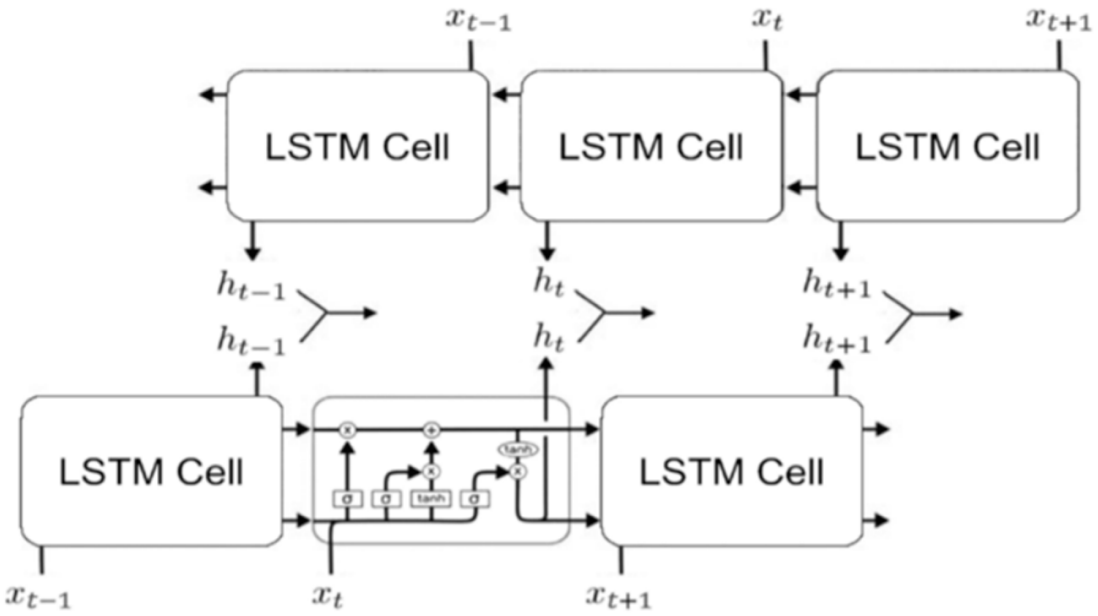

Figure 2: The Architecture of BiLSTM Cell

As can be seen, the BiLSTM divides the RNN's neurons in two directions. Therefore, this technique uses two-time directions and input data from the past and future of the current time frame. For both the univaritate LSTM and the BiLSTM, the parameters will be chosen based on the methodology explained in section 5.5.

5 Data and Methodology

In this chapter, the methodology used for forecasting the volatility of the Dutch Title Transfer Facility market between 2013 and 2023 will be described, as well as the data, pre-processing, and performance measurements will too be discussed and explained.

5.1 Data

As stated previously, the objective of this thesis is to estimate and compare econometric forecasting models against neural network models, based solely on the price. The thesis is conducted on the Dutch TTF daily spot prices. The data is consisting of End-Of-Day prices from Factset between the 2nd of September 2011 and the 17th of March 2023. The Dutch TTF prices have been chosen as it is used as a benchmark across all the hubs in Europe and is the most liquid and one of the supreme trading hubs across Europe, as previously established. There are a total of 2881 daily observations. In this thesis, the dataset will be split into two parts as follows:

- 3 full years (756 trading days) for validation and model tuning during training, i.e. Out-Of-Sample - **approx 26.24 percent**
- The remaining for training, i.e. In-Sample - **approx. 73.76 percent**

The training period will be from the 2nd of September 2011 to the 17th of March 2020, and the validation period will be from the 18th of March 2020 to the 17th of March 2023, providing a big enough period for the neural network models to learn and implement the complexity of the dataset, while also allowing the GARCH models to fit their estimation to the training set. To secure stationary in the data set, the log return of the closing prices is computed, further easing the estimation and further working with the model. The log returns will be computed in the following way:

$$r_t = ln(price_t) - ln(price_{t-1})$$

This will be done from $t = 1, ..., t = k$. Tables containing distribution plots of log returns and returns can be seen below:

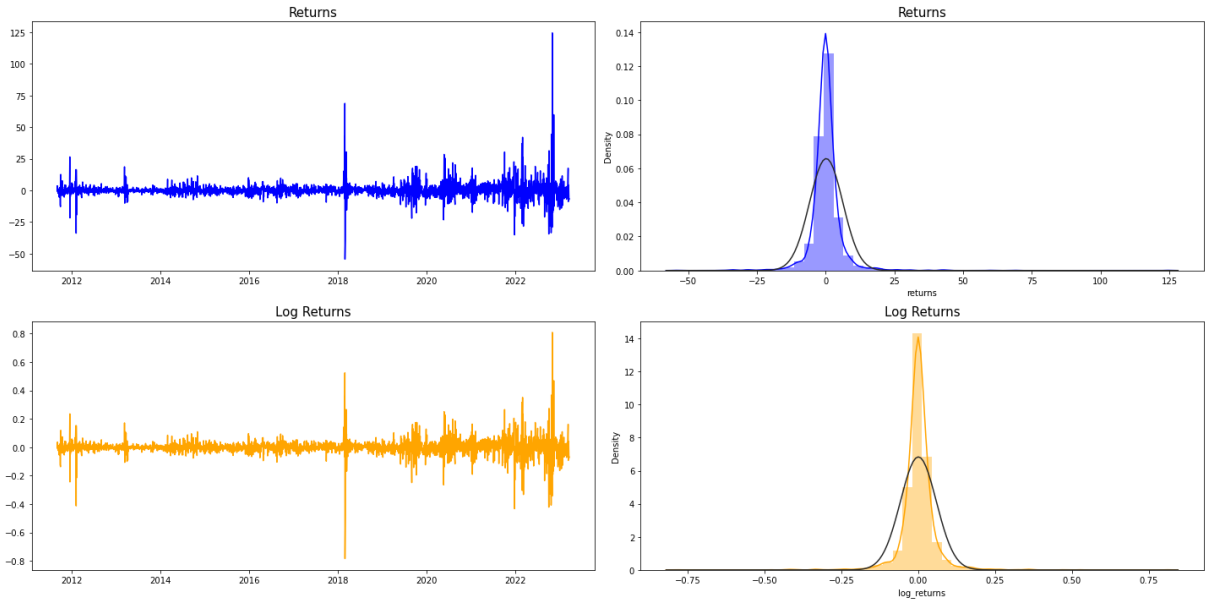

Figure 3: Distribution plots of returns and log(returns) of TTF gas spot prices

Both show signs of slight skewness and positive kurtosis. Furthermore, some degree of heteroskedasticity can also be detected whereby the following section will include the standard tests for pre-estimation.

5.2 Pre-Estimation of GARCH Model

Before fitting a GARCH model to financial time series data, it is essential to perform pre-estimation diagnostics to determine if the data exhibits certain characteristics required for GARCH modeling. These diagnostic tests assess whether the return series exhibits autocorrelation, conditional heteroskedasticity, stationarity, and normality.

5.2.1 Stationarity Test (Augmented Dickey-Fuller Test)

Firstly, the Augmented Dickey-Fuller (ADF) test, is performed for stationarity. This test is instrumental in ensuring the data is stationary, exhibiting constant mean, variance, and autocorrelation over time, which is an essential prerequisite for time series analysis (Dickey & Fuller, 1979). The mathematical form of the ADF test in its most general form can be written as:

$$\Delta Y_t = \alpha + \beta t + \gamma Y_{t-1} + \delta_1 \Delta Y_{t-1} + ... + \delta_p \Delta Y_{t-p} + \epsilon_t$$

Where Y_t is the time series, ΔY_t is the first difference of the time series, t is the time trend, α, β, γ and δ are parameters to be estimated, and ϵ_t is the error term.

The null hypothesis of the test is that $\gamma = 0$, indicating the presence of a unit root, i.e. the series is non-stationary. The alternative hypothesis is that $\gamma < 0$, indicating the series is stationary. Non-stationarity in the data can lead to misleading and untrustworthy model results, and in such cases, difference or other transformations may be employed to achieve stationarity. In this case, the '**adfuller**' package from the '**statsmodel**' library in Python is used on both the returns and log returns to compute the test statistic, critical values, and the p-value of each of the series. The p-values of the returns and log returns are 6.16147e-22 and 4.18044e-22 respectively. This indicates that both the series of log returns and returns are stationary.

5.2.2 Autocorrelation Test (Breuch-Godfrey test)

In time series analysis, particularly with GARCH models, we assume that there is no autocorrelation in the series after accounting for the conditional variance. To validate this assumption, the Breusch-Godfrey Serial Correlation LM test is performed on the residuals of the preliminary model. This test is more general than the Ljung-Box test, as it handles models with lagged dependent variables and higher-order autoregressive structures better. The hypothesis of the Breusch-Godfrey test can be expressed as follows:

$$H_0 = \textit{No autocorrelation}, p_1 = \cdots = p_p = 0$$

$$H_1 = \textit{At least one } p_p \textit{ is different from 0, autocorrelation}$$

With the test statistic being:

$$nR^2 \sim \chi^2_p$$

Where R^2 referes to the auxiliary regression, and n is the degree of freedom, where $n = t - p$.

The Breusch-Godfrey test assesses the null hypothesis that the error terms in a regression model are not autocorrelated up to a certain lag order. If we reject this null hypothesis, it indicates that there may still be some structure in the residuals of the model that hasn't been accounted for. This is important to check, as the presence of autocorrelation in the residuals can lead to inefficient parameter estimates and unreliable statistical tests(Breusch, 1978; Godfrey, 1978).

For this thesis, the '**acorr_breusch_godfrey**' test from the '**statsmodel**' library in Python is used to compute the test statistic and the corresponding p-value. As established in Appendix 1, the model used for this thesis is GARCH(1,1). The number of lags to include in the Breusch-Godfrey test can be selected based on a variety of factors, including the length of the time series and the characteristics of the data. However, a commonly employed rule of thumb suggests the number of lags is approximately the natural logarithmic function of the number of observations in the time series, hence lags= $ln2881 \approx 8$ (Tsay, 2010). Thus, the p-values from the Breusch-Godfrey test for GARCH(1,1) using 8 lag is 3.046e-14, indicating that there is strong evidence to reject the null hypothesis of no autocorrelation in the residuals. This suggests the presence of autocorrelation in the residuals of the GARCH(1,1) model. However, this is not necessarily an issue as the GARCH model implicitly assumes that there will be autocorrelation in the volatility of the data. Therefore, although the Breusch-Godfrey test indicates autocorrelation in the residuals, it does not invalidate the application of the GARCH model but rather informs us of the need to consider more complex model specifications or additional factors that may explain the autocorrelation in the residuals (Engle, 1982).

5.2.3 ARCH Effect Test (Engle's Test)

For testing the existence of autoregressive conditional heteroskedasticity (ARCH effects) in the time series, being a key assumption for GARCH models the Engle's Lagrange Multiplier test is used(Engle, 1982). Engle's LM test is a statistical test used to detect the presence of conditional heteroskedasticity in a time series. The test starts by estimating a simple autoregressive model:

$$y_t = \alpha + \beta \cdot y_{t_1} + \epsilon_t$$

Where y_t is the time series, α and β are parameters to be estimated, and ϵ_t is the error term. The next step is to square the residuals from this model, and regress them on constant and q-lagged values:

$$\epsilon_t^2 = \omega + \sum_{i=1}^{q} \gamma_i \cdot \epsilon_{t-i}^2 + v_t$$

Where ϵ_t^2 are the squared residuals from the first equation, ω and γ_i are parameters to be estimated, and v_t is the error term. The null hypothesis for the test is that the γ_i are all equal to zero, meaning no ARCH effect, against the alternative that at least one of

them is not zero. If the null hypothesis is rejected, this provides evidence of ARCH effects in the residuals of the model. For this thesis, the '**arch**' package provides functionality to this test, and thus will be utilized. The outcome of the ARCH effect test as well as the result of the estimation of lags for the empirical analysis can be seen in Appendix 1.

5.2.4 Normality Test (Jarque-Bera Test)

Even though GARCH models are flexible enough to handle non-normal distributions, a normality test will still be conducted as it helps in determining the distribution of the data and inform any necessary adjustments to the GARCH. For the purpose of this analysis, the Jarque-Bera (JB) test is used to assess the normality of the data distribution (Jarque & Bera, 1980). The JB test is built on the following hypotheses:

$$H_0 = \textit{Skewness and hurtosis match a normal distributino}$$

$$H_1 = \textit{Skewness and kurtosis does not match a normal distribution}$$

With the test statistic being:

$$JB = n \cdot [(\frac{S^2}{6}) + \frac{(K-3)^2}{24}]$$

Where n is the number of observations in the data set, S is the sample skewness, and K is the sample kurtosis. If the JB statistic is significantly different from 0, then the null hypothesis is rejected, indicating that the data do not have a normal distribution. In the case of this thesis, the '**jarque_bera**' package from the '**scipy**' library in Python is used for this test, computing the test statistic and p-value of both the returns and log returns. Both the returns and log returns have a p-value of 0, indicating that the data do not have a normal distribution. This was expected, as mentioned previously, time series data, especially natural gas data, are not normally distributed and are in fact negatively skewed, as natural gas volatility has heavy left tails and less heavy right tails. As mentioned at the beginning of this section, this is not an issue since GARCH models are able to handle non-normal distributions of the data.

5.3 Calculation of Realized Volatility

As the forecasting objective of the models is the realized volatility of the prices, we will discuss the calculation and choice of rolling interval period as well. As previously

mentioned, volatility measures the magnitude of price movements that certain financial instrument experiences over a certain period of time. As volatility is not directly observable, it has to be estimated. In this thesis, the realized volatility will be estimated as the square root of the sum of squares of log returns over a certain period divided by said period:

$$RV_t = \sqrt{\frac{\sum_{i=1}^{n} r_t^2}{n-1}}$$

Where r_t is the log(return) at time t and n is the interval window the realized volatility is calculated based on. When computing realized volatility, the choice of the interval for sampling returns plays a significant role in the statistical properties of the measure. The selection of the interval window length can affect both the noise level and the smoothing of the volatility estimate, thus the choice has to be a trade-off between noise reduction and capturing high-frequency changes in volatility. Choosing longer interval windows will reduce the impact of short-term noise and provide a more stable estimate of volatility over time, whereby it might also eliminate some genuine information about the changes in volatility, such as high-frequency changes and intra-day volatility due to smoothing. Choosing a short interval window instead allows volatility to respond faster to new information and capture more complex patterns, while it also allows for a lot of noise and "overfitting" of the volatility measure. Because of this, multiple interval windows will be chosen to get a more generalized picture. To determine which interval periods to choose, a visualization of realized volatility using the following interval windows will be made; 5-Day, 10-Day, 22-Day, 44-Day, 132-Day, and 252-Day intervals.

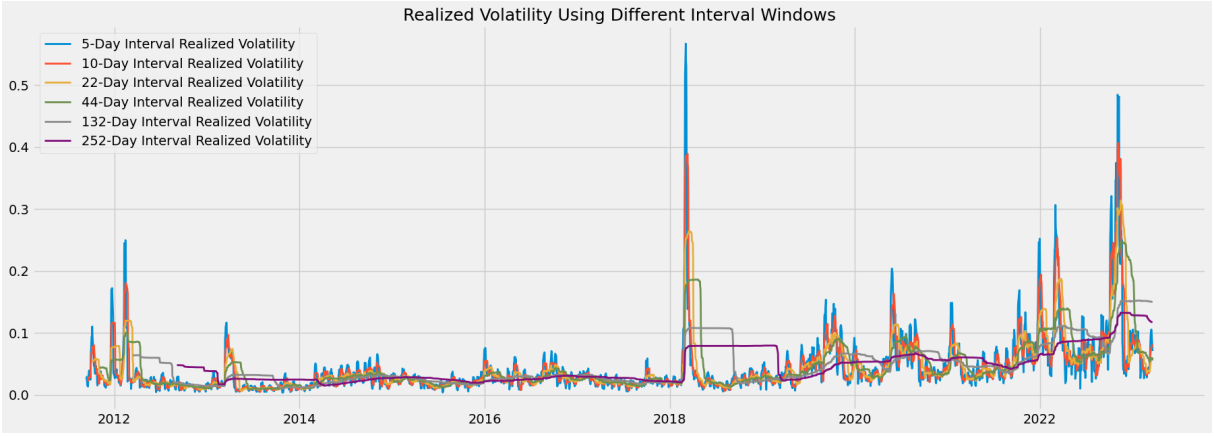

Figure 4: Realized Volatility Using Different Interval Windows

The volatility in the figure above is calculated based on the formula 5.3. It is apparent that each of the interval windows measures different magnitudes of volatility. We are interested in a lot of information without noise and more stationary data without smoothing away a lot of information. After the 22-Day interval, most of the information during periods of high volatility is smoothed away, whereby the 22-Day interval will be chosen as the highest amount. Even though the 5-Day interval is quite noisy, this thesis is dealing with gas spot data, whereby noise is expected, and depending on what the practical application of the forecasting models is for, a 5-Day interval might fit quite well. It might also be worth to investigate the middle of the two, to see if there is more balance between noise and smoothing. Therefore, the interval windows chosen for this thesis are the 5-Day, 10-Day, and 22-Day interval periods. I am aware that the choice of interval window will affect the "reality" and thus the basis that the models are built on. Therefore, it is acknowledged that each of the models across interval windows is based on its own "reality" whereby it will not be relevant to compare and assess across interval windows. Instead, an assessment for each of the interval windows will be made to better capture more generalized characteristics of each of the models.

5.4 Data Preprocessing

Before the implementation of our predictive models, it is crucial to carry out proper data preprocessing. This step is fundamental to ensure that our models can efficiently learn from the data, providing reliable and accurate forecasts. The first aspect of our data preprocessing involves scaling the data. Scaling of data is a standard procedure in machine learning applications and remains beneficial even when there is only a single input feature, thus affecting the stability and efficiency of the algorithms (Russell & Norvig, 2016). As this thesis are dealing with a variety of different models and algorithms, it would be better to normalize the volatility, so the forecasts generated by the different models will be standardized. For this purpose, we employ the **MinMaxScaler** from the **Scikit-Learn** library in Python. The MinMaxScaler is a tool for transforming features by scaling them to a specified range, in this case between 0 and 1. This technique helps mitigate the influence of outliers and ensures that all features have equal weight in our models, thereby helping to improve the performance of our predictions. Figure 5 is a visualization of the volatility distribution of the training dataset before and after scaling to see if there are any notable differences in scaling the volatility. As seen in the figure,

there are not any notable differences in the volatility distribution other than the values of the volatilize and the density, whereby it is said to now have affected the data by scaling it.

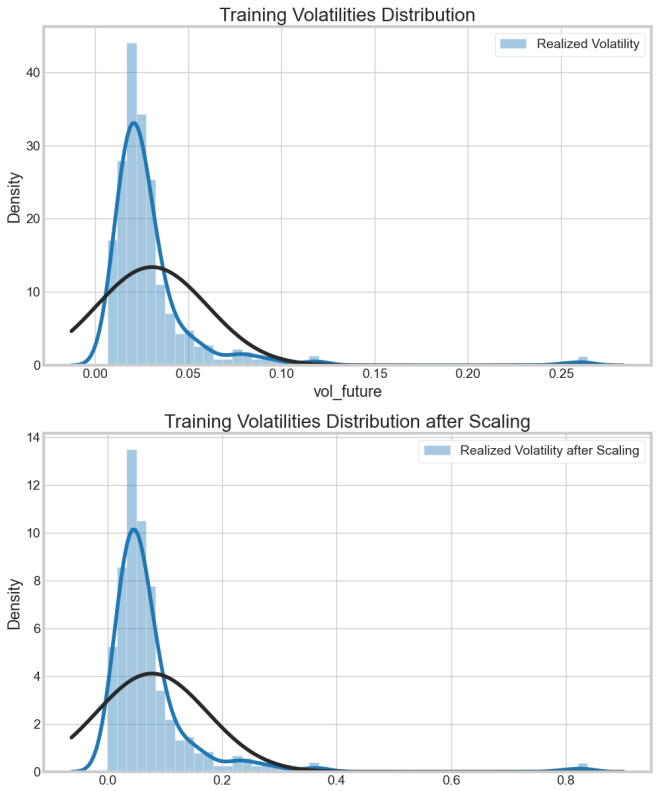

Figure 5: Volatility Distribution Before and After Scaling

As the GARCH models generally operate on return data, scaling the data pre-estimation would not make much sense. Instead, the GARCH models are scaled post-estimation, using the training data's conditional volatility arrays. As seen in Figure5, there is not a notable difference between the volatility distribution before and after scaling, whereby it is reasonable to assume that the essential volatility patterns have been preserved despite the change in scale. This approach aims to align the GARCH models' output with the scale of the ML models, enabling a more direct comparison of their predictive performances. However, It should be acknowledged that although this post-estimation scaling aligns the scale of the output, it does not necessarily alter the inherent properties of the GARCH models or the ML models. Therefore, the interpretation of their predictions must still reflect their underlying methodologies and objectives. As Figure 5 illustrates that the general shape and distribution of volatility are maintained post-scaling, thus indicating that while the scaling adjusts the range of the volatility values, it does not significantly distort the patterns of volatility over time, preserving the dynamic characteristics that

GARCH models aim to capture.

After scaling, we proceed to shift the data, creating our target variables for the forecasts. Shifting is the process of rearranging data points in a time series dataset to generate lagged variables, which are used as inputs to forecast future values. In this thesis, the data is shifted back by 1, 2, 3, 4, and 5 days, creating new variables that represent the values of the target variable from the previous 1 to 5 days. To put it more concretely, let us consider a single row in the time series for Day 6 (the original data), and the lagged values for Day 5 (D+1), Day 4 (D+2), Day 3 (D+3), Day 2 (D+4), and Day 1 (D+5). This structure is replicated across all the rows in our dataset. After shifting the data, the row will contain the variable values for Day 6 to accommodate the forecasts of D+1, D+2, D+3, D+4, and D+5. This approach allows for predicting the target variable's future values, providing valuable insights for training some of the forecasting models. The method of shifting and creating lagged variables is a standard practice in time-series forecasting and has been successfully used in many studies to improve the performance of forecasting models (Yan, 2010; Zhang, 2018).

5.5 Hyperparameter Selection for LSTM models

The selection of hyperparameters for LSTM and BiLSTM models has a significant impact on their learning capability and overall performance, thus this section will asses the chosen hyperparameters, and the reasoning behind them, supported by relevant literature and resources.

Batch Size:

Both models have a batch size of 64. Batch size refers to the number of samples that are propagated through the network at one time. According to Masters and Luschi (2018), generally, the best performance of NN models was by using a batch size between 16 and 64. Therefore, a batch size of 64 is a considered standard choice in LSTM and BiLSTM models, as it strikes a good balance between computational efficiency and the stochastic nature of the learning process.

Lookback period:

The univariate LSTM model has a lookback period of 10 days, and the BiLSTM model

uses a 30-day lookback period. The lookback period determines how many previous time steps the models should consider as input features when predicting the next time step. In practice, a longer lookback period allows the model to learn from longer sequences, but it also increases the complexity of the model and computation costs (Zhang et. al., 2015)

Number of Training Epochs:

Both models are trained for 200 epochs, where one epoch signifies one complete pass of the entire dataset through the network. Training for 200 epochs means that each sample in the dataset gets a chance to update the model's weights 200 times. This number of epochs can be effective to allow sufficient learning, while also preventing overfitting through excessive training.

Number of Units:

The LSTM layers in both models have 20 units, indicating the dimensional of the output in each layer. A moderate number of units, like 20, can help the model learn intricate temporal dependencies while managing the computational cost and risk overfitting.

It is important to note that this approach of choosing hyperparameters instead of optimizing each model, may limit the full potential of the neural network models as they are capable of optimizing these parameters to maximize performance. But, the primary objective of this thesis is not to pinpoint the best-performing model but to assess and compare machine learning models, specifically neural network models, performance to traditional econometric models, like GARCH models, in forecasting TTF spot price volatility. Thus, it is less relevant to find the optimal hyperparameters for each neural network model for each forecasting horizon under each interval window, and more important to be able to assess and compare the models fairly.

5.6 Forecast and Performance Measurements

After fitting the GARCH model, it's essential to evaluate its performance to ensure that it provides an adequate representation of the data. This will be done by estimating *Root Mean Squared Error* (RMSE) and the *Root Mean Square Percentage Error* (RMSPE) for all of the models. The best-performing representative, with RMSPE prioritized, will be included in the Diebold-Mariano Test to assess if the forecasting accuracy of the

model is significantly different or not.

The forecast accuracy of the two models is evaluated based on the Out-of-Sample validation, according to two performance measures: the RMSE and the RMSPE, with RMSPE prioritized as used in various other studies regarding machine learning techniques and financial time series (Song et al., 2022; Chen & Robert, 2022). They are computed as follows:

$$RMSPE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_i}{y_i}\right)^2}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

In this formula, y_i represents the actual value, $\hat{y}_i$ denotes the predicted value, and n is the total number of observations. The RMSPE is a measure of the differences between values predicted by a model and the values observed. It expresses this difference as a percentage, providing a relative measure of the error, which is particularly useful when we want to understand the error magnitude in the context of the actual values. Similarly, RMSE is an absolute measure of fit. It gives the standard deviation of the residuals, providing a measure of the prediction error. A lower RMSE value indicates a better fit to the data. By leveraging these two performance metrics, we aim to have a comprehensive understanding of our models' performance from both absolute and relative perspectives.

After conducting the analysis and finding the best representative for each of the models for the given interval window, a DM test will be conducted. The Diebold-Mariano test is a statistical test used to compare the accuracy of forecasting models. The null hypothesis in the Diebold-Mariano test states that two forecasting methods exhibit the same level of accuracy. The alternative suggests that the two methods differ in terms of accuracy (Harvey et al. (1997)). Mathematically, the hypothesizes can be expressed as:

$$H_0 : E(d_t) = 0 \quad \forall t$$

$$H_1 : E(d_t) \neq 0 \quad \forall t$$

Where $E(d_t)$ represents the expected value of the forecast error, and t denotes the time period. To calculate the forecast errors, the squared-error loss function is used, which is a common practice in the literature. The forecasted errors, denoted as d_t, are obtained by

taking the difference between the loss function applied to the forecast errors of the first model and the second model:

$$d_t = (y_i - \hat{y}_{i1})^2 - (y_i - \hat{y}_{i2})^2$$

Where y_i is the actual value, $\hat{y}_i$ denotes the predicted value of model 1, and $\hat{y}_{i2}$ denotes the predictive value of model 2. If both forecasting errors have the same accuracy, all d_t values will be zero, confirming that the null hypothesis holds true. The tables in the empirical analysis will contain a d - mean value, indicating the average forecasting error of the two models. A negative d - mean will indicate that the first model has a lower forecasting error than the second model, and vice versa with a positive d - mean value. Under the assumption that the DM test statistic follows a standard normal distribution, one would assume that there is a significant difference between the forecasts if $|DM| > Z_{crit}$, where z_{crit} is the two-tailed critical value for the standard normal distribution. For this thesis, a significance level of 5% is chosen.

5.7 Data Description

Before conducting the empirical analysis, an essential step will be to understand the nature and behavior of the underlying time series data, this being the realized volatility of the TTF gas spot prices across different interval windows. The investigation of the realized volatility across various interval windows was conducted in various ways. Initially, the summary statistic of each of the interval periods is included showing the mean, standard deviation, min, max, etc. Furthermore, the realized volatility for each interval window is plotted on a comparative scale for the entire duration of the data, while also specifically focusing on the most recent 365 days. This approach was undertaken to capture any trends or changes in volatility patterns. Subsequently, to uncover any potential seasonal trends and the influence of significant events that may appear as outliers, the realized volatility data were grouped by month and year. By examining the data in this manner, it is ensured that the analysis is more robust and enables a well-informed application of the models.

The following summary tables showcase the descriptive statistics for the entire dataset and the validation period, across three different interval windows: 5-day, 10-day, and 22-day.

Descriptive Statistic (All Data)			
	5-Day Interval	10-Day Interval	22-Day Interval
Mean	0.068595	0.091185	0.115876
Std	0.088049	0.112706	0.137280
Min	0.000000	0.000000	0.00000
25%	0.025162	0.033399	0.042062
50%	0.043272	0.055795	0.070933
75%	0.077109	0.108705	0.142011
Max	1.000000	1.000000	1.000000

Table 1: Descriptive Statistic for the Entire Dataset

Descriptive Statistic (Validation)			
	5-Day Interval	10-Day Interval	22-Day Interval
Mean	0.128587	0.172718	0.22116
Std	0.116698	0.146251	0.173427
Min	0.011397	0.038947	0.051774
25%	0.059481	0.083012	0.110568
50%	0.096033	0.131647	0.179600
75%	0.148303	0.197693	0.244375
Max	0.853518	1.00000	1.00000

Table 2: Descriptive Statistic for the Validation Period

Table 1 focuses on the entire dataset. It reveals that the mean volatility gradually increases with the length of the interval window, ranging from approximately 0.069 in the 5-day window to 0.116 in the 22-day window. This pattern is also observed in standard deviation and the 75th percentile, suggesting increased variability and dispersion in volatility as the interval window expands. As is apparent, the Min and Max for the entire dataset are 0 and 1, respectively. The reason for this is that the data is scaled based on the entire dataset on a scale from 0 to 1, whereby this is apparent in Table 1. Furthermore, Table 2 provides statistics for the validation period. Compared to the entire dataset, the validation period demonstrates a notably higher mean and standard deviation across all interval windows. This indicates higher average volatility and greater dispersion during the validation period, meaning the validation period incurred higher levels of volatility than the training period. It is worth noting that the minimum volatility is no longer zero, reflecting a higher floor for volatility values in this period. Ultimately, these tables highlight the distinct statistical characteristics of gas spot price volatility under different periods and interval windows, setting the context for the subsequent model comparison and analysis. The following figures will display the realized volatility, with respect to each of the interval windows, and also grouped by month and year.

As reflected in Figure 6, the 5-day interval window depicts a more volatile set of volatility, while the 22-day interval window exhibits a more stable but persistent realized volatility. This discrepancy becomes particularly prominent when focusing on the last 365 days, with the realized volatility for the 5-day interval window being markedly more noisy and susceptible to sudden spikes than the longer interval periods. Some noteworthy observations were apparent from the spikes at the beginning of 2018 and the overall increase in volatility levels around 2020, peaking in 2022. These spikes correspond with significant events in the energy sector, such as severe weather conditions in the first quarter of 2018, leading to low gas storage levels and panic across the gas markets, leading to the TTF spot prices closing around €545/MWh (Betley, 2018).

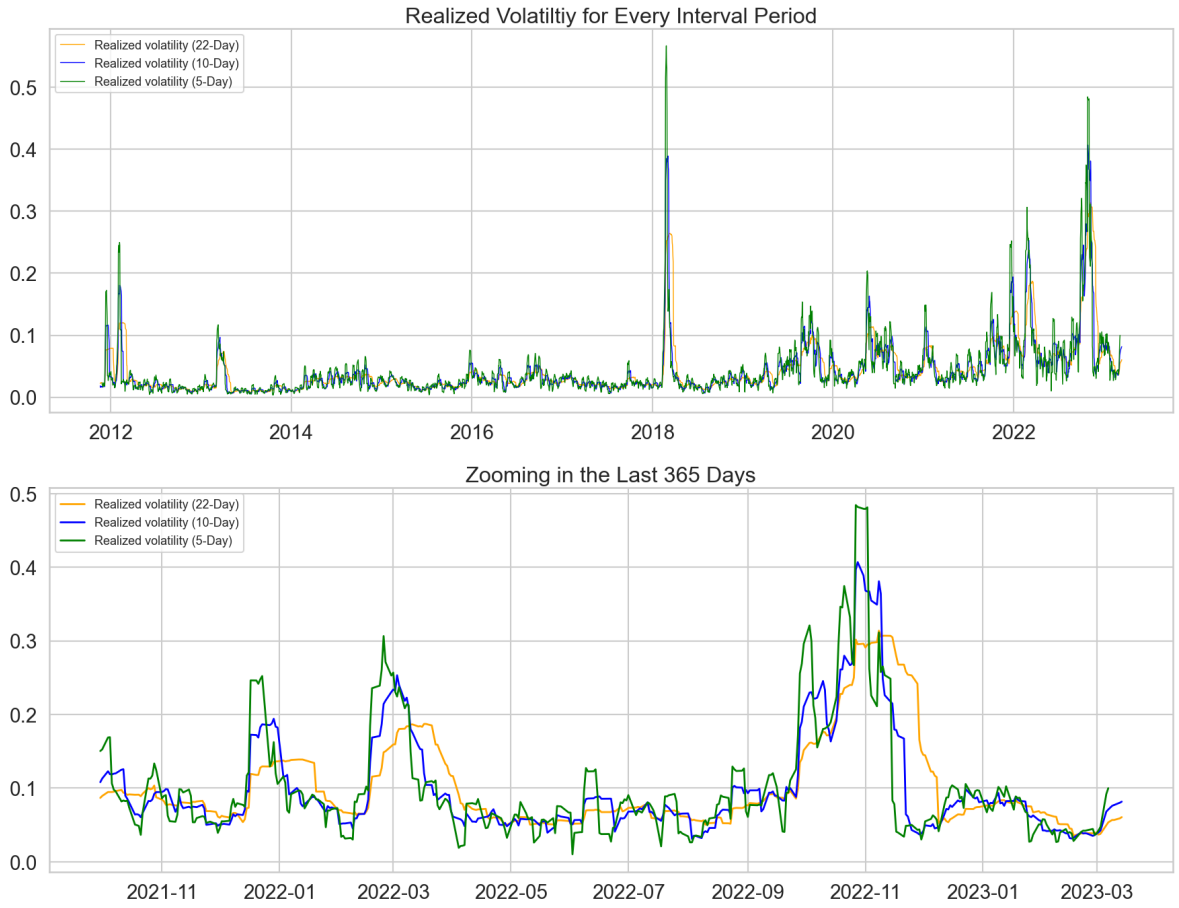

Figure 6: Realized Volatility Using Different Interval Windows

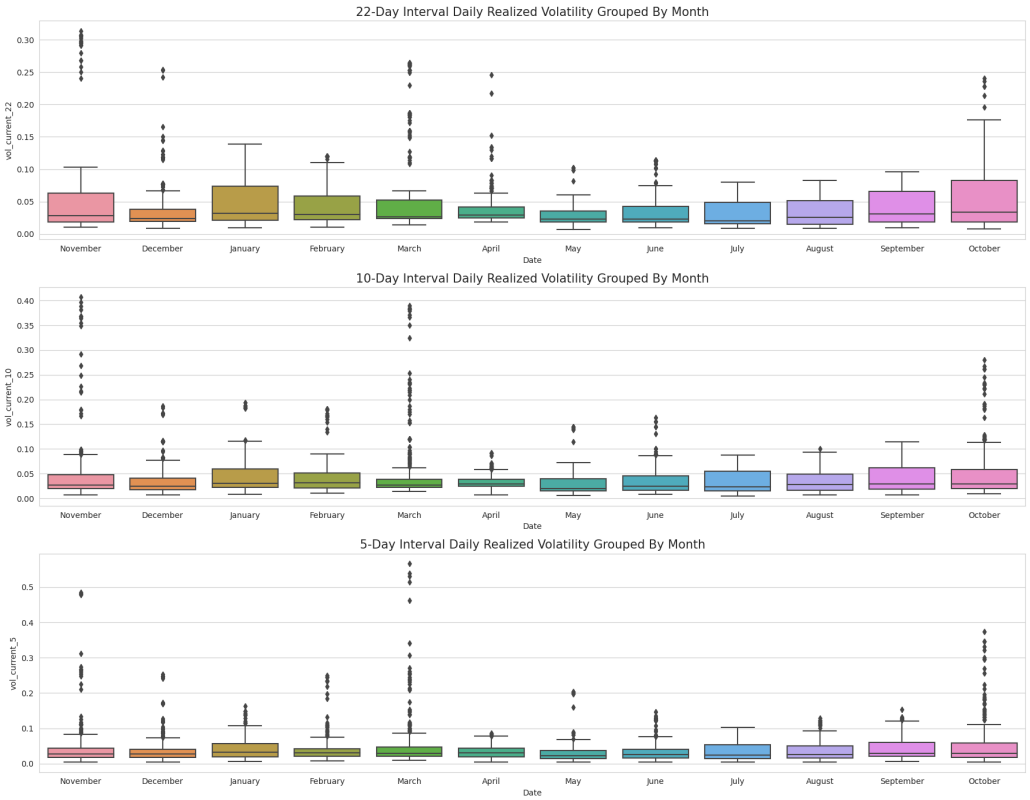

Figure 7: Daily Volatility Grouped By Month

Further analysis of volatility patterns divided by month and year revealed that the months from October to March are significantly more volatile across all interval periods. This aligns with the fact that during these months, the demand for gas, and therefore gas prices and associated volatility, are typically higher Notably, global events seemed to have a tangible impact on the volatility levels. For instance, the global lockdown due to Covid-19 led to panic and speculation in the markets, and the escalation of the war in Ukraine in late February 2022 significantly influenced the volatility in certain periods and could be explanatory factors for the outliers in these months.

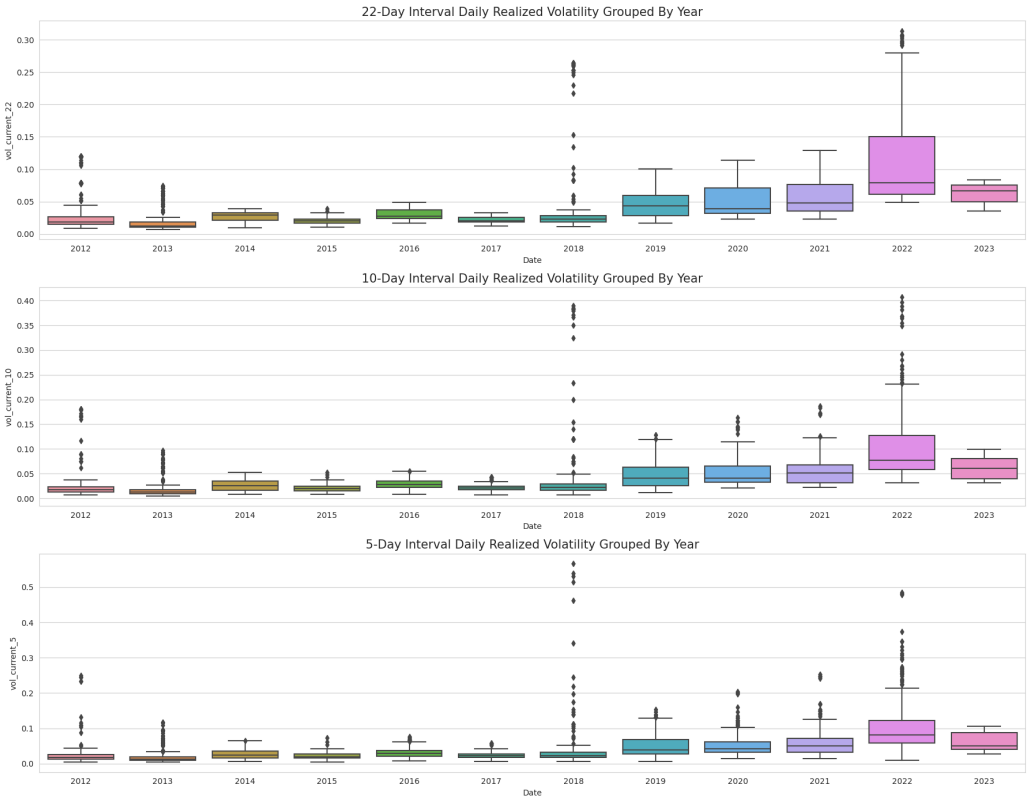

Figure 8: Daily Volatility Grouped By Year

As apparent in Figure 8, the period 2011 has been removed as the dataset only contained about 2.5 months of data in that period. whereby it was deemed that it was not a rightful representation of the entire year. Regardless, a year-on-year comparison of the volatility data revealed interesting patterns. While the period from 2014 to 2017 exhibited relatively low volatility levels, the years 2012 and 2013 witnessed some outliers across all interval windows, however, these were relatively minor compared to the years to come. Especially 2018 experienced high levels of volatility due to cold winters and low storage levels, as explained above. Years post-2018 witnessed a sharp increase in volatility, perhaps due to the increased volatility in 2018, resulting in increased speculation and trading volume in the gas market. Particularly, the year 2022 stands out as a year of extreme volatility, likely due to several disruptive factors influencing global energy markets. The year 2022, however, stands out for its extreme volatility. This year witnessed the mean of the boxplot surpass the high of the body across all interval windows. This substantial increase in volatility, with outliers of daily volatility reaching about 0.3, 0.41, and 0.49 for the 22-day, 10-day, and 5-day intervals respectively, underscores 2022 as an unusually volatile year, likely driven by several disruptive events in the gas market. As for the current year, 2023, the volatility levels seem to have regressed to levels akin to 2021.

Nevertheless, as with 2011, the data for 2023 should be interpreted cautiously, considering that we are only partway through the year and as previously noted, the month of October typically sees the highest volatility. This preliminary analysis of the volatility data has yielded valuable insights into the behavioral dynamics of the realized volatility of TTF gas spot prices and will serve as a foundational stepping stone for the subsequent stages of our empirical analysis.

6 Empirical Analysis and Discussion

As previously established, market volatility is a fundamental characteristic of financial markets, directly impacting the risks and rewards associated with various investment opportunities. The ability to accurately forecast market volatility is critical for the development of effective trading strategies, risk management, and the optimization of returns. Traditional econometric models such as the GARCH and its variants, including the GJR-GARCH, have been employed for this purpose for many years. However, the emergence of machine learning techniques, particularly deep learning models like LSTM networks, has opened up new possibilities in the field of financial forecasting.

Thus the objective of this empirical analysis is to compare and evaluate the performance of a diverse range of forecasting models, encompassing both conventional approaches and advanced machine learning techniques, the various GARCH models and LSTM models respectively. These evaluations and comparisons will be done through multiple interval windows, calculating the realized volatility on a 5, 10, and 22-day basis. Ultimately, the evaluation of these models within each of these interval periods will be evaluated using the DM test, with the best-performing model representing the entire forecasting period for that model to see if the accuracy of the forecasting models is significantly different.

6.1 5-Day Interval Window

6.1.1 Naive Model

One of the common characteristics of volatility is that it tends to be autocorrelated and clustering in the short term. This characteristic can be leveraged to construct a simple model that forecasts future volatility solely based on the immediate prior daily volatility. In this case, the average daily volatility used will be the most recent interval window, being the past 5 days, as predictions for the next 1 to 5 days, thus using current volatility at time t for predicting future volatility at time $t+1$. Such an approach can be referred to as a "naive" model, given its reliance on a single feature and lack of sophisticated methodology. Nonetheless, it can serve as a baseline for comparison with more complex models, such as GARCH or LSTM.

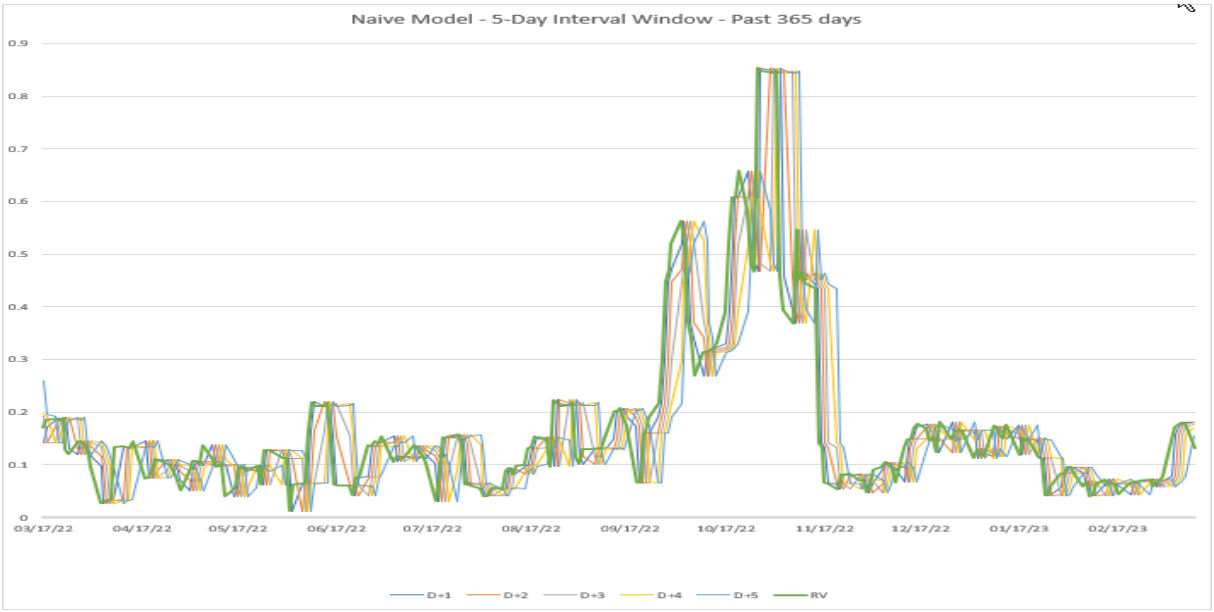

Figure 9: Naive Forecasting Model Using 5-Day Interval Window

As expected, the forecasted volatility is equal to the realized volatility, lagged by the forecasting period, also indicating that the D+1 naive forecasting model performs the best. The relative and absolute performance measurements can be seen below:

Naive (5-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.457299	0.040747
D+2	0.573209	0.059380
D+3	0.721430	0.073663
D+4	0.836840	0.085386
D+5	0.958083	0.096539

Table 3: Forecasting performance for naive model (5-Day interval period)

As seen in the figure above, the D+1 naive forecasting was performing the best, both on a relative and an absolute scale. Thus indicating that the D+1 forecasting model is the best of the naive forecasting horizons, using a 5-day interval window for forecasting realized volatility. However, an RMSPE of 0.457299, meaning the model's prediction. on average, deviates from the actual value by about 45.73%, thus a better model will be needed. However, this yields a good benchmark for other models.

6.1.2 GARCH

In this section, attention is turned to the empirical analysis of the GARCH (1,1) model. This model, as detailed in Appendix 1, encompasses a single lag of past returns and a single lag of past residuals. The primary focus of this section will be to assess the model's ability to capture the volatility dynamics present in the data and to evaluate its forecasting performance in comparison to the other models considered in this thesis.

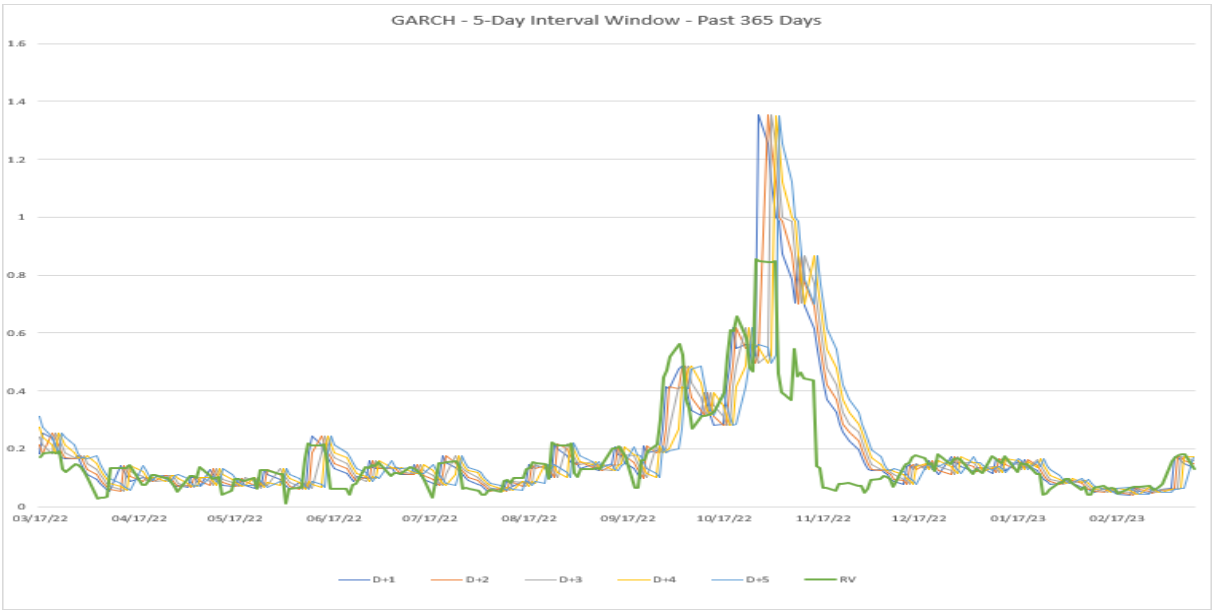

Figure 10: GARCH Forecasting Model Using 5-Day Interval Window

Empirical observations indicate that the GARCH is fitting quite well with the data during periods of low volatility. However, it shows a tendency to overestimate sudden spikes of volatility and does not seem to be able to readjust when volatility decreases again. This is quite an issue, as the relevance of volatility forecasting reaches its highest in periods of high volatility, as this means higher spreads and the possibility for traders to profit. Thus, the GARCH model might not be sufficient in forecasting volatility in the short term, which also undermines the model's performance measurements.

The performance measurements for the GARCH model reaffirm the identified limitations. From the perspective of RMSPE and RMSE, the GARCH model is inferior to the naive model across all forecasting horizons. More specifically with and RMSPE of 0.549963 against 0.457299 at the D+1 forecasting horizon, it underperforms the naive model in terms of accuracy. This inferiority is consistent across all forecasting horizons

GARCH (5-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.549963	0.066246
D+2	0.668229	0.078886
D+3	0.803795	0.092040
D+4	0.956777	0.105703
D+5	1.134150	0.119586

Table 4: Forecasting performance for GARCH(1,1) model (5-Day interval window)

from D+1 to D+5 with RMSPE and RMSE values worsening as forecasting horizons increase. This evidence indicates that the GARCH model for the 5-Day interval window is not sufficient. It's performance is surpassed by the naive model, and it's inability to forecasting volatility in periods of high volatility is a significant drawback.

The basic GARCH model is based on the assumption that the residuals and the mean returns are normally distributed. However, this is often not the case when dealing with financial time series data as they do not follow a normal distribution, and is more likely to be extreme positive or negative values that deviate significantly from the mean. In Figure 11, a table of the estimated standardized residuals from the GARCH model is shown.

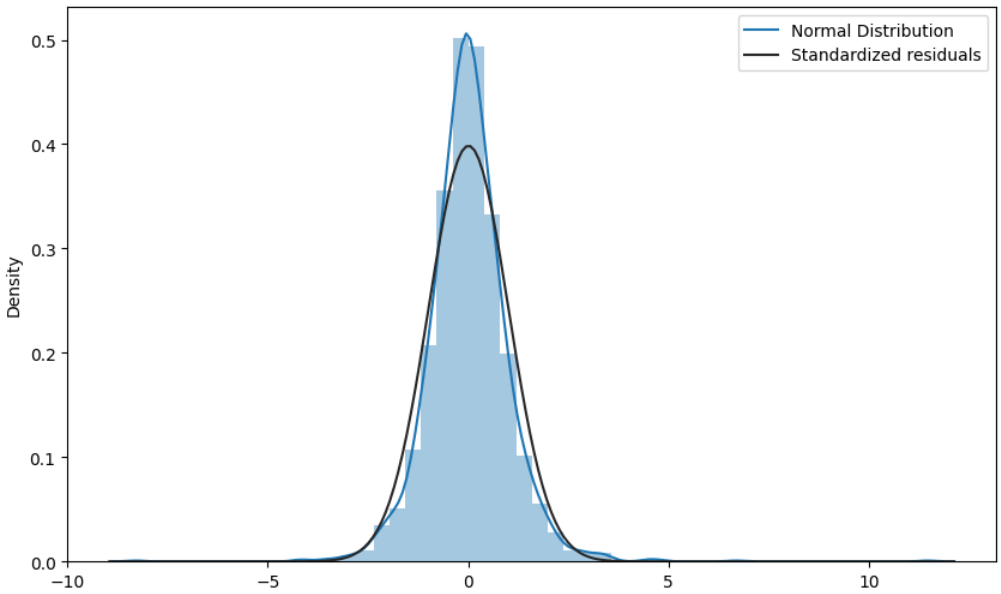

Figure 11: GARCH Residual Distribution for 5-Day D+5 Forecasting

A slightly skewed distribution of residuals, can be seen, indicating that the volatility

of the TTF spot prices is impacted asymmetric and that negative impacts might affect the volatility more than the positive ones. Therefore a GARCH model that takes asymmetric shocks into account would be worth investigating.

6.1.3 GJR-GARCH

The GJR-GARCH is a variation of GARCH that takes asymmetric shock responses into account, contrary to the standard GARCH that presumes that both positive and negative volatility news have similar impact on volatility.

The GJR-GARCH will follow the same methodology as the standard GARCH, using the same lag parameters ($p = 1, q = 1$). It should be noted that even though the presented example only highlights asymmetry in the 5-Day interval, D+5 forecasting GARCH model, the GJR-GARCH model will be estimated for all of the interval windows and forecasting horizons to ensure comparability and to help indicate whether machine learning models genuinely outperform traditional econometric models.

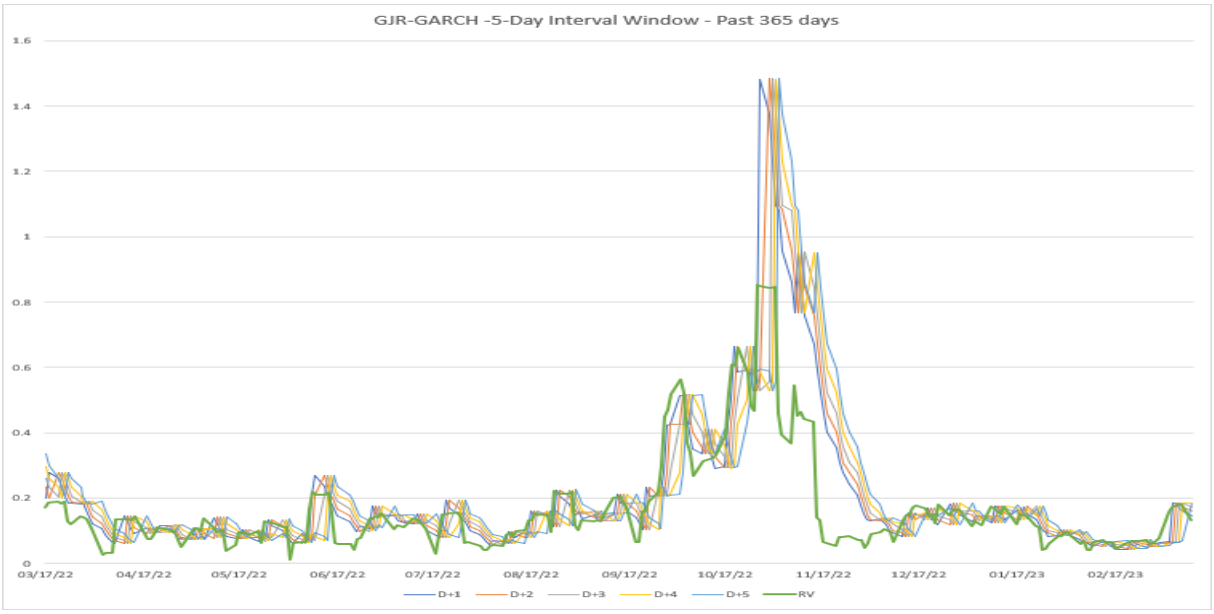

Figure 12: GJR-GARCH Forecasting Model Using 5-Day Interval Window

As with the standard GARCH, the GJR-GARCH struggles with estimating the magnitude of sudden increases in volatility. However, one notable distinction of the GJR-GARCH model is it seems to adapt more rapidly to negative readjustments, indicating this model may offer some improvements over the standard GARCH model. Despite this, the similarity between the standard GARCH and the GJR-GARCH models implies that accounting for asymmetric shocks might not drastically impact volatility forecasting for

the 5-Day interval window.

GJR-GARCH (5-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.614748	0.076936
D+2	0.745684	0.089382
D+3	0.891385	0.102170
D+4	1.053470	0.115478
D+5	1.24190	0.129325

Table 5: Forecasting performance for GJR-GARCH(1,1) model (5-Day interval window)

Performance measures of the GJR-GARCH model, unfortunately, do not corroborate the initial visual impressions of superiority. The RMSPE for the 4-day and 5-day forecasts exceeds 1, indicating that, on average, the deviation of the forecasted values from the actual values exceeds 100% of the actual values. This discrepancy could be attributed to the noise data present in the 5-day rolling window period, as discussed in section 5, or the model's inability to capture sudden spikes in volatility effectively. Consequently, the empirical evidence suggests that both the GARCH and GJR-GARCH models may not be adequate for forecasting volatility of the TTF spot prices, at least for the 5-Day interval window. Thus, more advanced models with potential capabilities to forecast volatility more accurately would be required.

6.1.4 Univariate LSTM

The following sections will shift the focus from the econometric models to the exploration of machine learning models, specifically this section will detail the empirical study of the univariate LSTM model.

Visually, the LSTM model for the 5-Day interval window fits quite well with the validated data, especially the D+1 and D+2 forecasts. This suggests a strong alignment between the forecasted values and the validation data of the LSTM model, which indicates smaller forecasting errors, and thus implying a more accurate model compared to the GARCH models. However, there are still some slight lags, as were the case with the other models, but seemingly this model seems to perform better than the GARCH models.

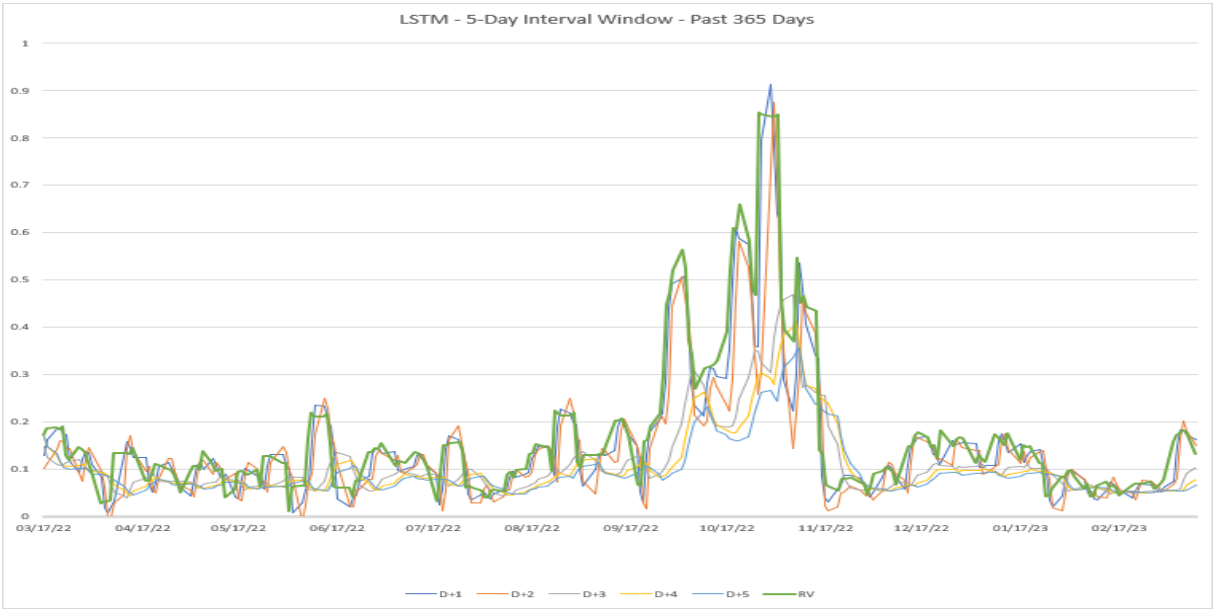

Figure 13: LSTM Forecasting Model Using 5-Day Interval Window

LSTM (5-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.4020998	0.076936
D+2	0.4743965	0.089382
D+3	0.53846	0.102170
D+4	0.558603	0.115478
D+5	0.559719	0.129325

Table 6: Forecasting performance for LSTM model (5-Day interval window)

From a performance perspective, the LSTM model has shown improvement across all forecasting horizons compared to previous models. Notably, the LSTM model is the first of the models to outperform the naive forecasting model, which historically has been quite tricky. Furthermore, it is also the first model to demonstrate consistency across all of the forecasting horizons, suggesting it is a more reliable model for forecasting across all days compared to models like the GARCH model, which showed considerable deviation from the lowest RMSPE of 0.549963 to the highest being 1.13415. Given the promising results of this univariate LSTM model, it would be interesting to investigate whether augmenting the model with additional layers could yield further performance improvements. Therefore, the subsequent section will delve into the bidirectional LSTM

model.

6.1.5 Bidirectional LSTM

Following the investigation of the univariate LSTM model, this section will explore the empirical analysis of a more complex machine learning architecture - the Bidirectional Long Short-Term Memory (BiLSTM) model. The BiLSTM model utilizes the same hyperparameters as the univariate LSTM model, however, the look-back period is extended to 22 days instead of the previously used 14 days. This is explained by the fact that the complexity of the BiLSTM potentially enables better capture of hidden layers, thus justifying an extended look-back period.

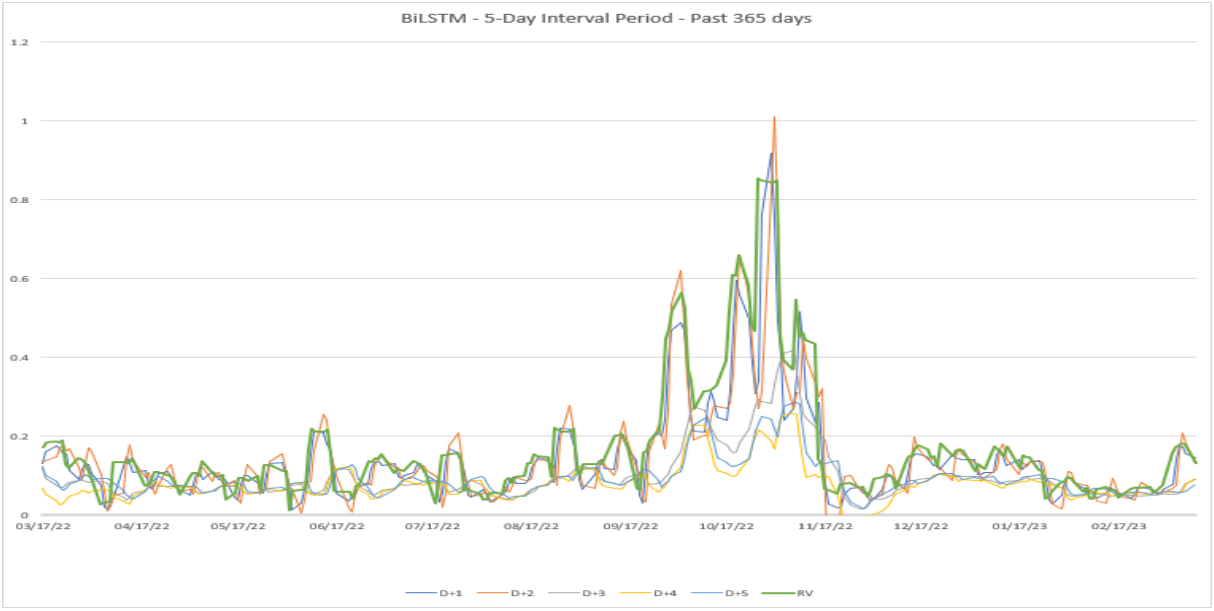

Figure 14: BiLSTM Forecasting Model Using 5-Day Interval Window

From a visual perspective, the BiLSTM model appears to align better with the realized volatility on short forecasting horizons. However, it continues to underestimate volatility during D+3, D+4, and D+5 forecasting horizons in periods of high volatility. Additionally, it seems to overestimate sudden increases or spikes of volatility but is accommodating quite well to the readjustments. This behavior was not observed in either of the GARCH models.

Despite the increased complexity of the BiLSTM model, its performance metrics show only a marginal improvement over the univariate LSTM model. Both the relative and absolute performance measures are superior at the lower forecasting horizons. In contrast,

BiLSTM (5-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.380186	0.042888
D+2	0.482661	0.064569
D+3	0.525121	0.084864
D+4	0.559510	0.112117
D+5	0.5564064	0.107328

Table 7: Forecasting performance for BiLSTM model (5-Day interval window)

the univariate LSTM model is performing better at D+4 and D+5 forecasting. This suggests that the LSTM models are strong contenders, but not necessarily a superior choice, as the outperformance might not be significant.

After conducting the analysis of each model and comparing them across forecasting horizons, an investigation into whether the best-performing model, in this instance the BiLSTM model, significantly outperforms the most optimal representations of each other model, is conducted. This comparison is performed using the Diebold-Mariano test, and the results are as follows:

DM Test (5-Day interval)				
Model 1	Model 2	d - mean	P-Value	Result
BiLSTM (D+1)	Naive (D+1)	0.0003968	0.2708327	Insignificant
BiLSTM (D+1)	GARCH (D+1)	-0.002332	0.292616	Insignificant
BiLSTM (D+1)	GJR-GARCH (D+1)	-0.003862	0.276478	Insignificant
BiLSTM (D+1)	LSTM (D+1)	0.0002924	0.231563	Insignificant

Table 8: 5-Day Interval Window DM Test Results (5% Significance Level)

Notably, the GARCH models have a slightly higher forecasting error compared to the BiLSTM model, measured by the negative d - mean, compared to the naive and LSTM models. However, all four p-values are greater than the significance level of 0.05. Thus, the null hypothesis cannot be rejected for any comparison, suggesting that no significant difference exists in the 5-day forecast accuracy between the BiLSTM model and other models (LSTM, Naive, GARCH, and GJR-GARCH). Thus, according to the DM test

results and when utilizing a 5-day interval window for volatility forecasting, the BiLSTM model does not provide forecasts significantly different from those of the LSTM, Naive, GARCH, or GJR-GARCH models. However, it is important to note that this does not imply that all models are of equal quality or performance, but rather that their forecasting performance does not differ significantly in the context of this test and dataset.

To summarize, the BiLSTM model with the specified hyperparameters appears to provide the most superior performance in terms of both RMSPE and RMSE among the models analyzed in this thesis. The univariate LSTM model, although marginally underperforming compared to the BiLSTM model, remains a strong contender, outperforming the naive, GARCH, and GJR-GARCH models. Interestingly, the naive forecasting model also performs well, especially considering its simplicity, outperforming both GARCH models. The following section will extend this investigation by applying the same methodology to the models using a 10-Day interval window, as opposed to the 5-Day interval window used so far, in an attempt to understand if the findings remain consistent across different volatility calculation periods.

6.2 10-Day Interval Window

6.2.1 Naive Model

The analysis of volatility forecasting now continues, expanding the interval window to a 10-day period. This choice might smooth out the data a bit, but it could also introduce more stationarity. In this scenario, the model uses the average daily volatility from the past 10 days as a predictor for the upcoming 1 to 5 days. This is a shift from the previous naive model, which used a 5-day interval window.

The realized volatility is quite similar to that seen with the 5-day forecasting window, though there are fewer minor spikes, and, the volatility is persisting more in certain levels. As expected, the forecasted volatility from the naive model mirrors the realized volatility delayed by the forecast duration. This once again indicates a strong performance of the D+1 naive forecasting model.

As is apparent in the table above, the D+1 forecasting horizon is superior to other forecasting horizons using the naive forecasting model, as measured in both relative and absolute terms. The RMSPE of 0.1612004 shows that the D+1 forecasting model, on

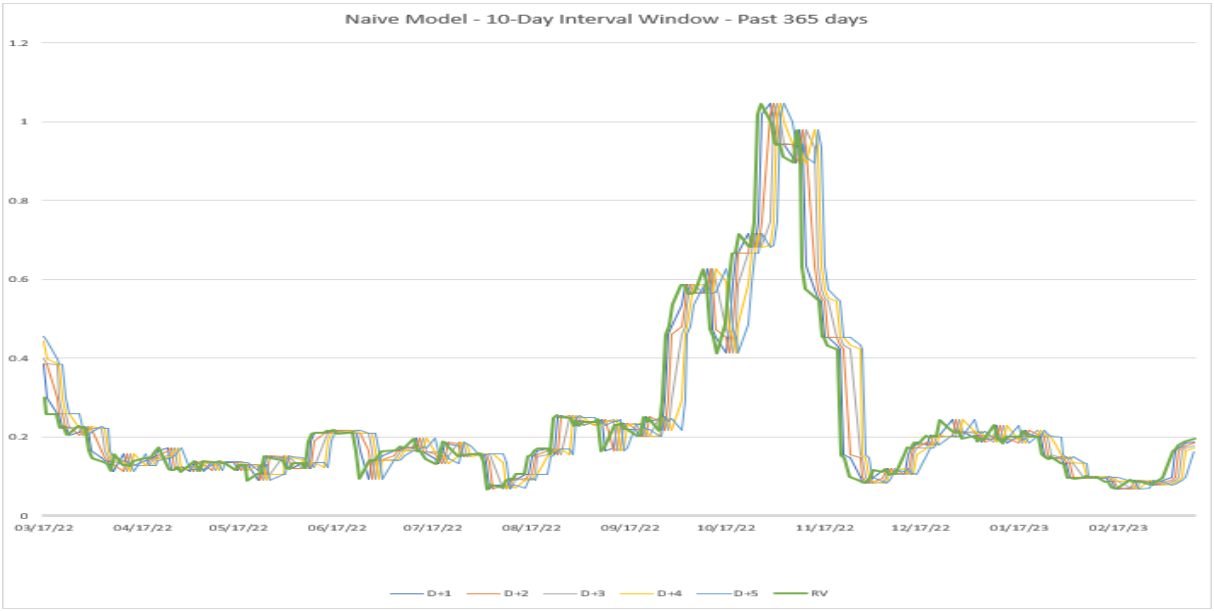

Figure 15: Naive Forecasting Model Using 10-Day Interval Window

Naive (10-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.1612004	0.031912
D+2	0.2392257	0.049037
D+3	0.3174454	0.062645
D+4	0.3892040	0.074761
D+5	0.4602220	0.086365

Table 9: Forecasting performance for Naive model (10-Day interval window)

average, deviates from the actual value by about 16.12% making it a robust benchmark for the upcoming models.

6.2.2 GARCH

The GARCH model applied to the previously investigated interval window showed less-than-optimal performance compared to other forecasting models. This shortfall could be due to its inability to adjust quickly to abrupt changes in volatility within the noisy short-term data of the 5-Day interval window. Therefore, it would be interesting to see how the model performs when the realized volatility is smoothed out for sudden spikes.

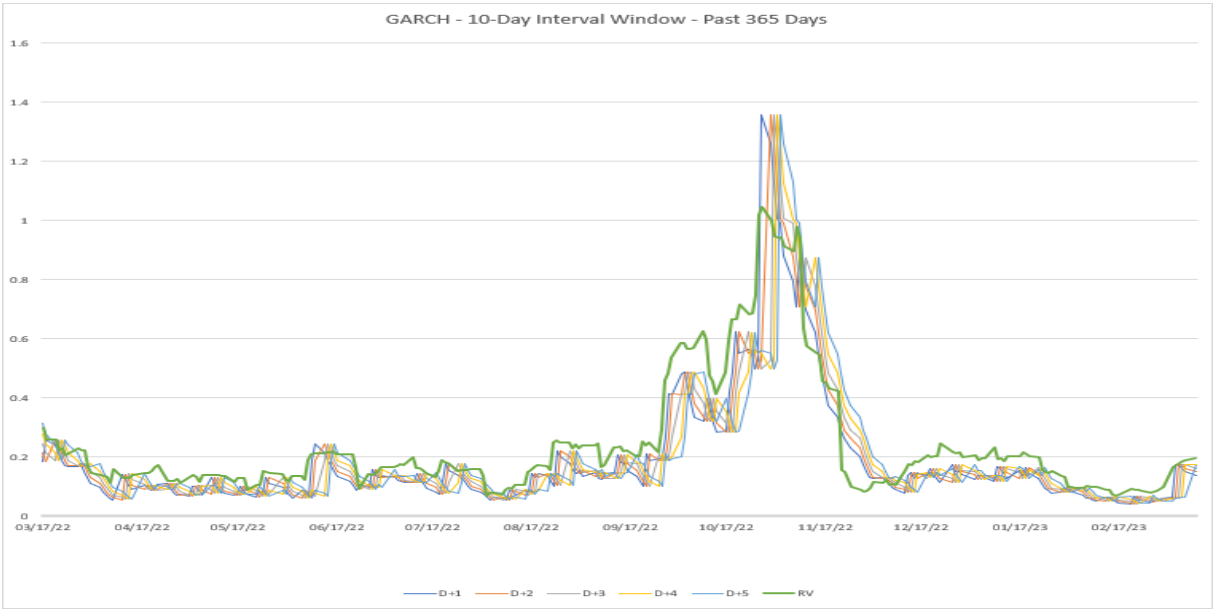

Figure 16: GARCH Forecasting Model Using 10-Day Interval Window

In the GARCH model for the 10-day forecasting period, some of the same tendencies are present, including the model's difficulty in correctly forecasting the level of volatility during periods of increased volatility. However, visually, the GARCH model appears to fit the realized volatility better, even though it still lags behind the realized volatility, similar to the naive model, and it tends to underestimate the realized volatility for most of the period. The performance measurements are presented below:

GARCH (10-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.348477	0.068119
D+2	0.356609	0.074478
D+3	0.370430	0.080801
D+4	0.390130	0.087449
D+5	0.415455	0.094898

Table 10: Forecasting performance for GARCH(1,1) model (10-Day interval period)

As was apparent from figure 10, the D+1 forecasting horizon was superior, but it still could not outperform the naive forecasting model. Nonetheless, the GARCH model outperforms the GARCH model using the 5-Day interval window based on RMSPE, but not RMSE. It is crucial to note that even though this GARCH model seems superior to the 5-Day interval GARCH model, they cannot be directly compared. However, it is quite

interesting that the GARCH model did not manage to forecast volatility on the shorter interval window due to factors such as noisy data and sudden movements in volatility. Therefore, it would be interesting to see if a longer interval period might lead to improved performance of the GARCH model.

6.2.3 GJR-GARCH

The GJR-GARCH model which used a 5-Day interval window, previously showed an inability to adjust to large movements in volatility, whereby the most recent GARCH model showed some improvement in the less noisy data. Thus, it would be interesting to see if these tendencies persist with this variation of GARCH.

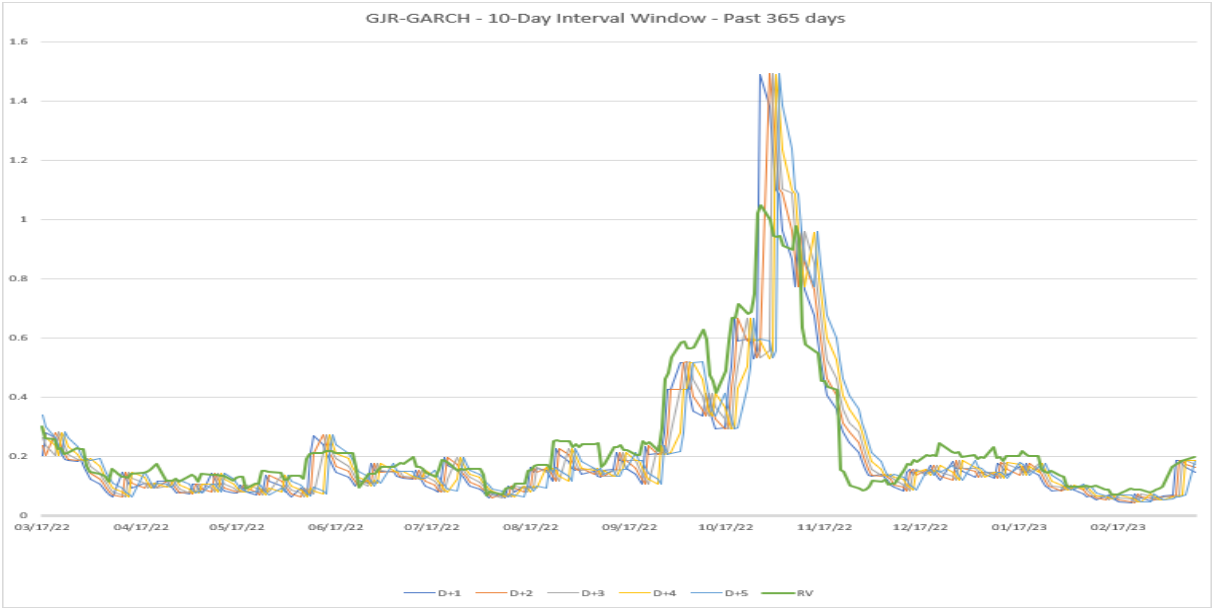

Figure 17: GJR-GARCH Forecasting Model Using 10-Day Interval Window

As with the standard GARCH model, the GJR-GARCH forecasts seem to fit better to the realized volatility. However, the forecasted volatility of the GJR-GARCH model still overestimates the high level of volatility that occurred around late 2022. Furthermore, the model slightly underestimates the volatility, especially in on the longer forecasting horizons, from D+3 to D+5.

As can be seen from the table above, the GJR-GARCH model performs best on the shorter forecasting horizon, even marginally outperforming the standard GARCH model. As with the standard GARCH model, the GJR-GARCH model did not deviate as much between the forecasting horizons, making these models more stable when using a 10-Day

GJR-GARCH (10-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.320008	0.064780
D+2	0.333030	0.072275
D+3	0.353840	0.079620
D+4	0.381437	0.087092
D+5	0.415713	0.095396

Table 11: Forecasting performance for GJR-GARCH (1,1) model (10-Day interval window)

interval window than using a 5-Day interval window. However, neither of the GARCH models outperforms the best-performing forecasting model from the naive forecasting model, although they do outperform the D+4 and D+5 forecasting horizon on a relative measure, achieving an RMSPE of 0.381437 and 0.415713 respectively. This outcome indicates that the GARCH models may be superior to the naive model over longer forecasting horizons.

6.2.4 Univariate LSTM

The machine learning models outperformed the traditional GARCH models, as well as the naive forecasting model, under the 5-day interval window. It would be interesting to see if the same tendency persists here. The same hyperparameters and methodology will be used for the models under the 5-day interval window will be used for the following models.

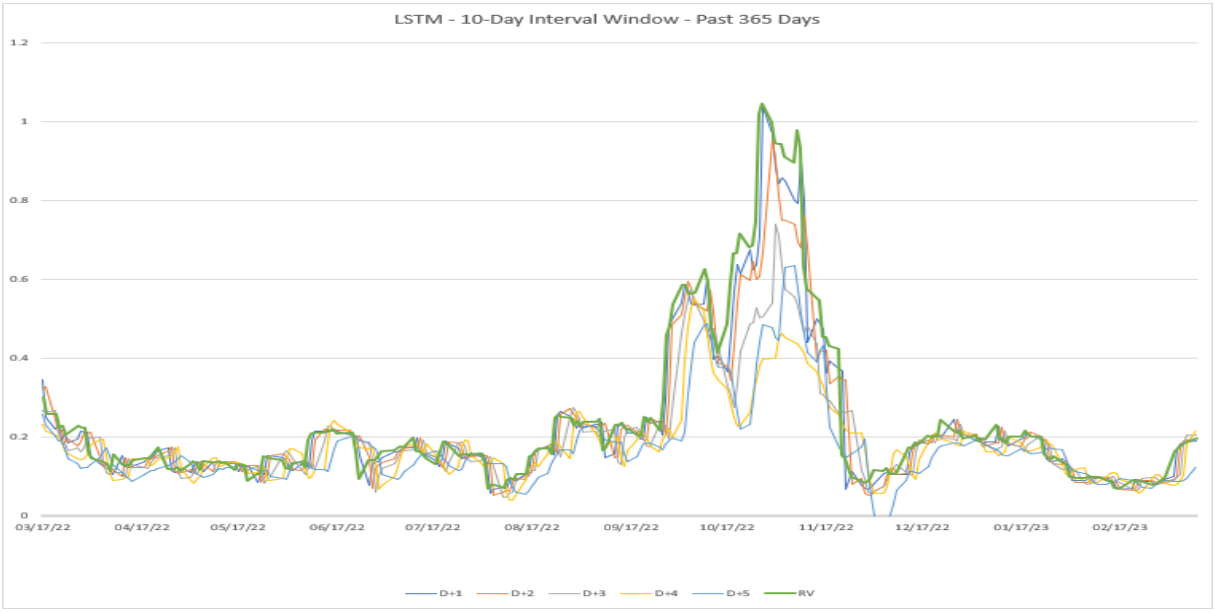

Figure 18: LSTM Forecasting Model Using 10-Day Interval Window

Some of the same tendencies are present for this interval window, similar to the previous one. The LSTM model still underestimates the realized volatility, especially under the longer forecasting horizons, and lags slightly behind the realized volatility. Visually, it appears that the D+3 to D+5 forecasting is not able to capture the magnitude of the higher levels of volatility but adjusts quite well when the volatility is readjusted back to lower levels.

LSTM (10-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.320008	0.064780
D+2	0.333030	0.072275
D+3	0.353840	0.079620
D+4	0.381437	0.087092
D+5	0.415713	0.095396

Table 12: Forecasting performance for LSTM (10-Day interval window)

Once again, the univariate LSTM model is outperforming the remaining models on a relative scale, but based on the RMSE, the naive model is still slightly better as a forecasting model. However, since RMSPE is prioritized, the LSTM model is said to be the best model so far. It has an RMSPE of 0.153993 and 0.348934 for the D+1 and D+5, respectively. This result means that, on average, the model's forecast of the volatility of

D+1 deviates by approximately 15.39% from the actual value and the volatility of D+5 deviates approximately 34.89% from the actual value. This model is also quite reliable on a relative scale as the deviation from the best-performing forecasting horizon to the worst is only about 19 percentage points.

6.2.5 Bidirectional LSTM

The BiLSTM was the best-performing model for the previous interval window. The outperformance of the machine learning models seems to persist for the univariate LSTM model, so it will be interesting to see if the same can be observed from the BiLSTM model using a 10-Day interval window.

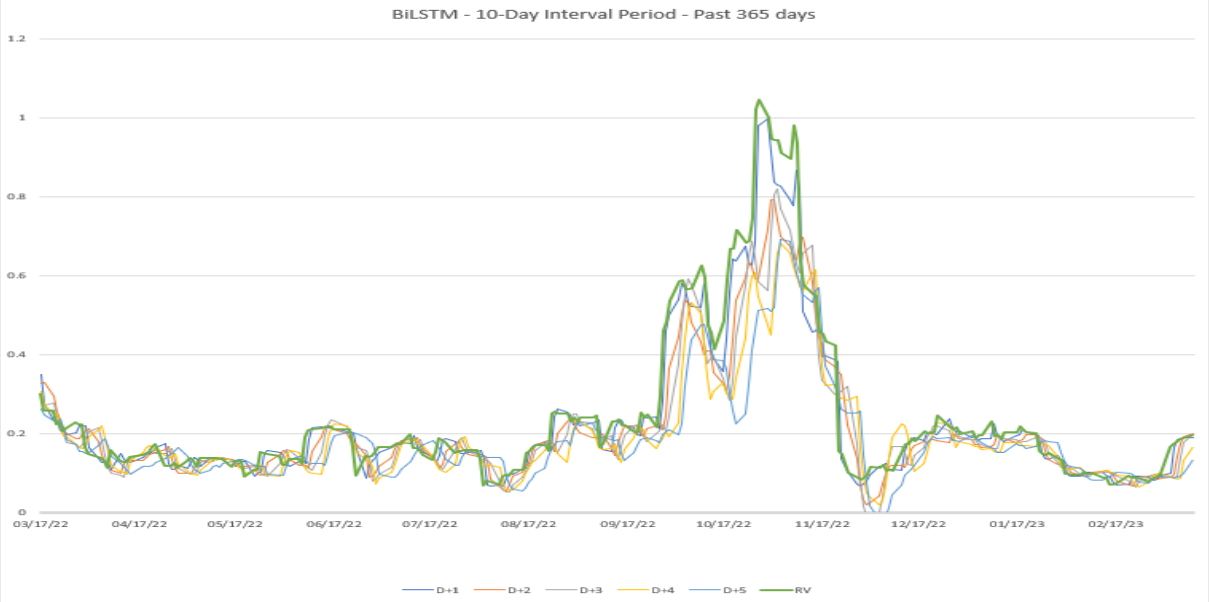

Figure 19: BiLSTM Forecasting Model Using 10-Day Interval Window

The visualization of the univariate LSTM and the BiLSTM are quite similar. The BiLSTM model also underestimates the realized volatility and lags slightly behind, indicating that this model is fully capable of forecasting the volatility accurately. However, like the univariate LSTM model, the BiLSTM forecasting model adjusts quite well to sudden decreases in volatility when it adjusts back to the lower levels, indicating a performance similar to the univariate LSTM for the 10-Day interval window.

BiLSTM (10-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.154771	0.033662
D+2	0.231008	0.059832
D+3	0.272629	0.066022
D+4	0.302503	0.082325
D+5	0.365213	0.092315

Table 13: Forecasting performance for BiLSTM (10-Day interval window)

The BiLSTM model is slightly outperformed by the univariate LSTM model, which is expected given their similarities across the forecasting horizon. The univariate LSTM model seems to be lightly better at fitting the volatility at the spikes, but other than that, the two models are visually the same. Therefore, whether the outperformance of the univariate LSTM model is in fact significant will be investigated in the test below:

DM Test (10-Day interval)				
Model 1	Model 2	d - mean	P-Value	Result
LSTM (D+1)	Naive (D+1)	0.0000432	0.50248	Insignificant
LSTM (D+1)	GARCH (D+1)	-0.003579	0.02756	Significant
LSTM (D+1)	GJR-GARCH (D+1)	-0.003135	0.07328	Insignificant
LSTM (D+1)	BiLSTM (D+1)	-0.000072	0.28940	Insignificant

Table 14: 10-Day Interval Window DM Test Results (5% Significance Level)

Analyzing the results of the DM test, it is apparent that the p-value for the comparison between the LSTM and the naive, GJR-GARCH and BiLSTM is above the significance level of 5%, indicating that the LSTM fails to significantly provide a different forecast than the remaining models. However, the DM test between the LSTM and standard GARCH model shows a p-value of 0.2756, suggesting a significant difference in the performance of the two models and indicating the LSTM model is in fact outperforming the traditional GARCH model using a 10-Day interval window. The difference in forecasting can also be seen in the d-mean, as it indicates that the GARCH models have higher forecasting error than the LSTM models, as was the case for the 5-Day interval as well. Furthermore, the

GJR-GARCH is only slightly insignificant with a p-value of 0.07328. If a significance level of 10% or even 7.5% were chosen, this comparison would also reject the null hypothesis, further indicating the outperformance of the GARCH models. The remaining interval period, the 22-Day interval period, will see if the findings of the previous two interval periods can be elaborated.

6.3 22-Day Interval Window

6.3.1 Naive Model

For the remaining interval period, the realized volatility is based on a longer time frame, whereby it would be expected the result in more stationary and smooth out volatility. Therefore, it is expected that some of the models will perform quite well in forecasting volatility, given some of the tendencies that have been observed so far. As with the other interval windows, the naive model will use the average daily volatility of the recent 22 days as the predictor for the coming 5 days.

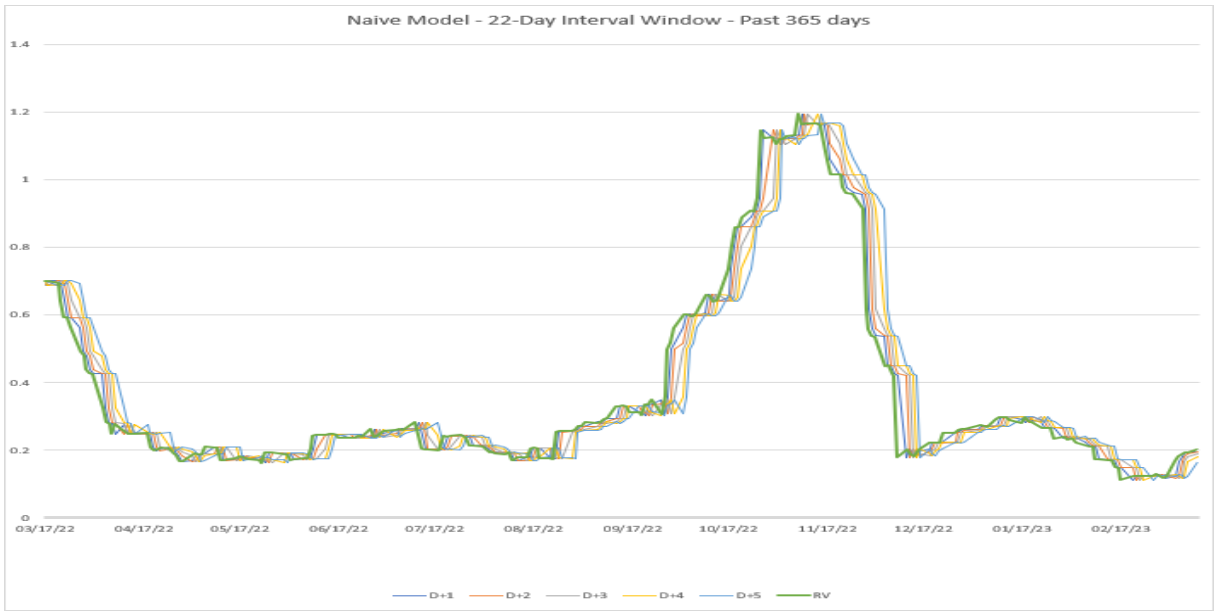

Figure 20: Naive Forecasting Model Using 22-Day Interval Window

With a 22-Day interval window, the realized volatility is more smoothed out, with less noise before and after the significant increase at the end of 2022, with the large spike being more substantial and persistent. As with the previous naive models, the forecasted volatility mirrors the realized volatility, lagging by the duration of the forecast horizon.

Naive (22-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.09381	0.02598
D+2	0.13722	0.04020
D+3	0.17511	0.05261
D+4	0.20980	0.06430
D+5	0.24631	0.07570

Table 15: Forecasting performance for Naive model (22-Day interval window)

The naive model for the D+1 forecast achieves an RMSPE below 0.1 which is quite good. It is, however, expected given the realized volatility is smoothed out, and the naive model for the D+1 forecasting horizon traces the realized volatility quite closely. Regardless, it is quite notable and will serve as a difficult benchmark to surpass.

6.3.2 GARCH

In the previous sections, it was apparent that the GARCH model's performance improved when forecasting was based on a 10-Day interval instead of a 5-Day interval window. Therefore, it would be interesting to see if the tendency persists when the interval window is further increased.

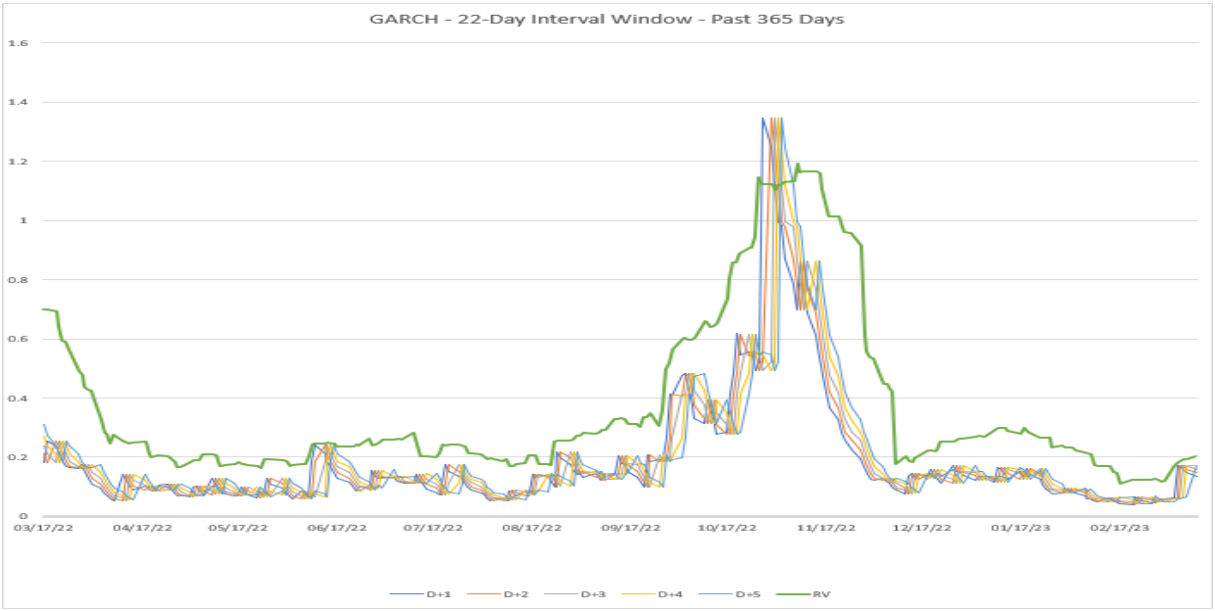

Figure 21: GARCH Forecasting Model Using 22-Day Interval Window

Visually, it is apparent that the GARCH is considerably underestimating the volatility for the entire period. It is not capable of capturing the correct level of volatility consistently at any given level during the last year of the validation period. Furthermore, consistent with observations made throughout the analysis, the GARCH model's forecast lags the realized volatility by the duration of the given forecasting period. The only instance where the forecasted volatility is above or close to the realized volatility is when the large increase occurs in late 2022, similar to the other GARCH models. This does not indicate a good performance of the standard GARCH model.

GARCH (22-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.54116	0.171958
D+2	0.53862	0.169685
D+3	0.53627	0.167839
D+4	0.54188	0.166396
D+5	0.53142	0.165405

Table 16: Forecasting performance for GARCH model (22-Day interval window)

Although results from the previous interval window cannot be directly compared, it is apparent that the performance of the GARCH model has declined using a 22-Day interval window, both on a relative and absolute scale. Interestingly, for the first time, a long-term forecasting horizon (D+5) has outperformed the shorter-term forecasting horizons. Although the performance of the model is still not satisfying, the D+5 model will represent the standard GARCH model during the DM test.

6.3.3 GJR-GARCH

The GJR-GARCH model has demonstrated certain strengths in previous forecasts, including its ability to adjust to the asymmetry of volatility distribution. However, this has also proven to be one of the flaws of the model, as it seems to overestimate the increases in volatility, and is unable to readjust quickly enough to provide a significant forecast. Thus, it is interesting to see if the tendency remains the same for the 22-Day interval window.

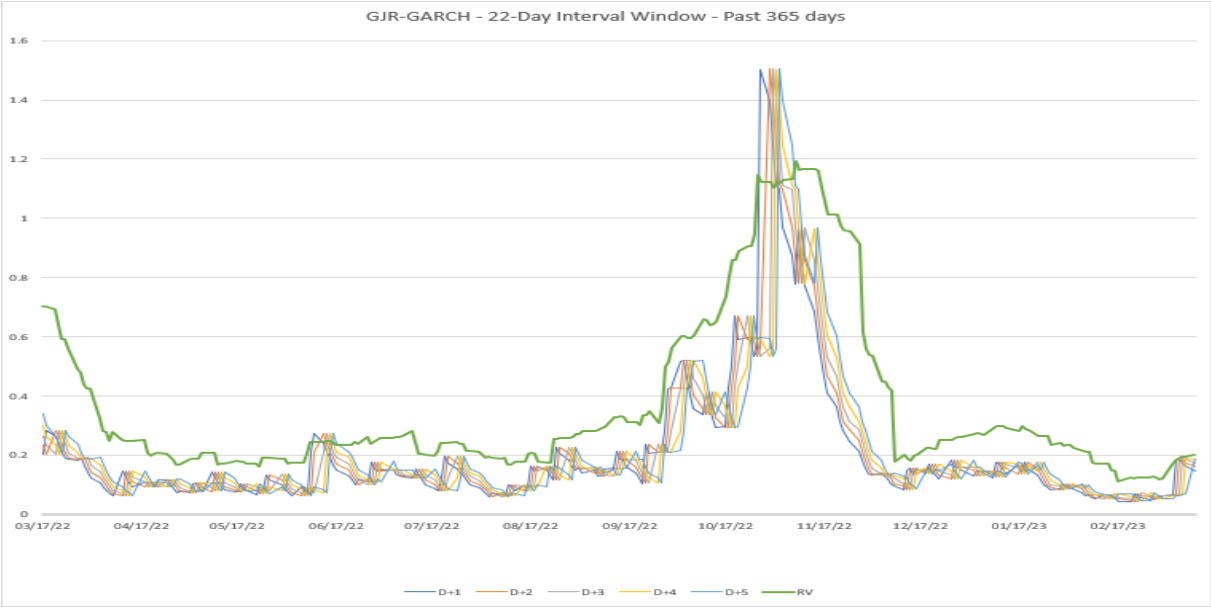

Figure 22: GJR-GARCH Forecasting Model Using 22-Day Interval Window

At first glance, the tendencies appear to be similar to those observed with the standard GARCH. The GJR-GARCH model using the 22-Day interval window is underestimating the forecasted volatility, and even though it has the ability to adjust, it seems like it is consistently underestimating, with very little adjustment for asymmetry. Like the standard GARCH and the previous GJR-GARCH models, the GJR-GARCH for this interval period is overestimating the sudden increase in volatility, and almost immediately after readjusting to underestimate for the remainder of the period. Visually, it does not indicate that this model can outperform the simple naive forecasting model.

GJR-GARCH (22-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.50735	0.160447
D+2	0.50411	0.157635
D+3	0.50224	0.155730
D+4	0.50078	0.154335
D+5	0.49818	0.153268

Table 17: Forecasting performance for GJR-GARCH model (22-Day interval window)

Similar to the standard GARCH, the GJR-GARCH performs better at forecasting the D+5 horizon than the D+1. The GJR-GARCH model does not outperform the

benchmark but does outperform the standard GARCH model measure both on RMSPE being 0.49818 against 0.531419 and RMSE being 0.153268 against 0.165405.

6.3.4 Univariate LSTM

The machine learning models have consistently outperformed other models across previous interval windows, while the GARCH models have yet to beat the naive benchmark. This naive benchmark is indeed a significant reference point in this context, with an RMSPE below 0.1. It would therefore be interesting to see if either the univariate LSTM or the BiLSTM forecasting model can outperform the benchmark.

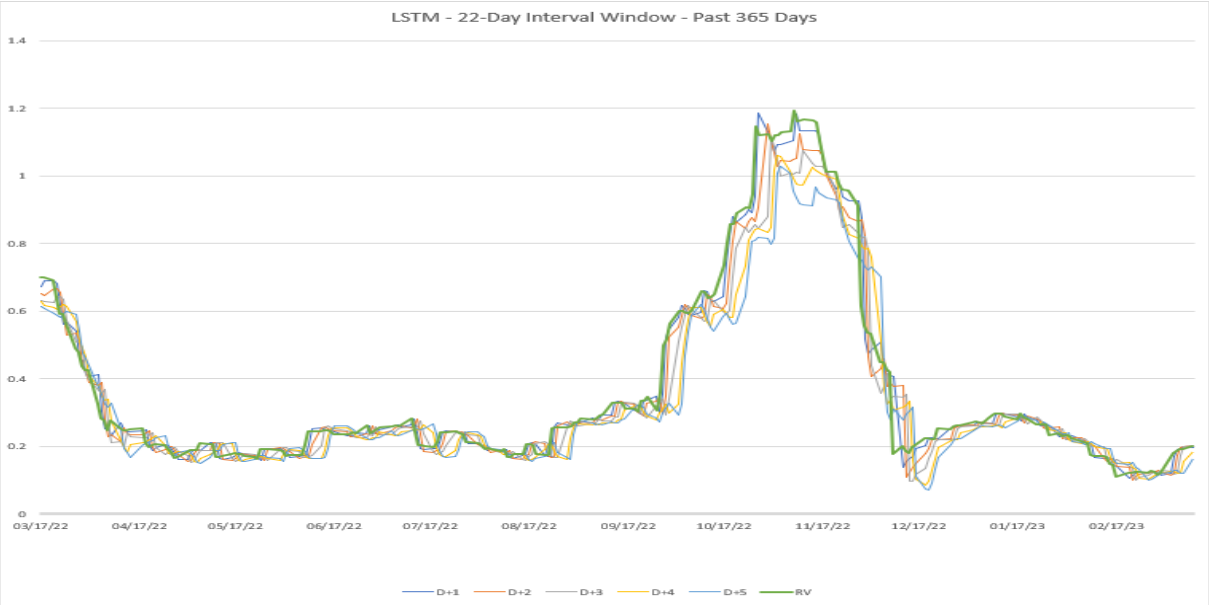

Figure 23: LSTM Forecasting Model Using 22-Day Interval Window

The LSTM model appears to follow the realized volatility quite close, and at times even predicts significant volatility increases before they occur in the realized data, which is not yet seen. Like the models used in previous interval windows, the LSTM model continues to underestimate the longer time horizons, although not as significantly as the previous models. This could be due to the data being more smoothed out using a 22-Day interval window, thereby reducing noise in the data. Nonetheless, this LSTM model seems to forecast volatility that aligns well with the realized volatility during the validation period.

The performance of the LSTM model is quite impressive, even outperforming the naive forecasting model. The univariate LSTM model for the D+1 forecasting horizon has an RMSPE of 0.09199, indicating that the model, on average, produces forecasts that deviate

LSTM (22-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.09199	0.025336
D+2	0.12850	0.039103
D+3	0.15555	0.050384
D+4	0.17916	0.062019
D+5	0.19973	0.071524

Table 18: Forecasting performance for LSTM model (22-Day interval window)

9.19% from the actual value. This result is quite good and the best among the models tested so far. However, as we will discuss later, a lower RMSPE does not automatically signify a superior forecasting model, as it is easier to forecast realized volatility when it is smoother and more stationary. Nevertheless, it would be interesting to see if the increased complexity of the BiLSTM model will prove a better model for forecasting volatility using a 22-Day interval window.

6.3.5 Bidirectional LSTM

As observed above, the univariate LSTM model managed to surpass both the naive benchmark and the two GARCH models, thereby indicating the advantages that machine learning models can bring to forecasting tasks. Nevertheless, to reach a more comprehensive assessment, the BiLSTM model still needs to be analyzed.

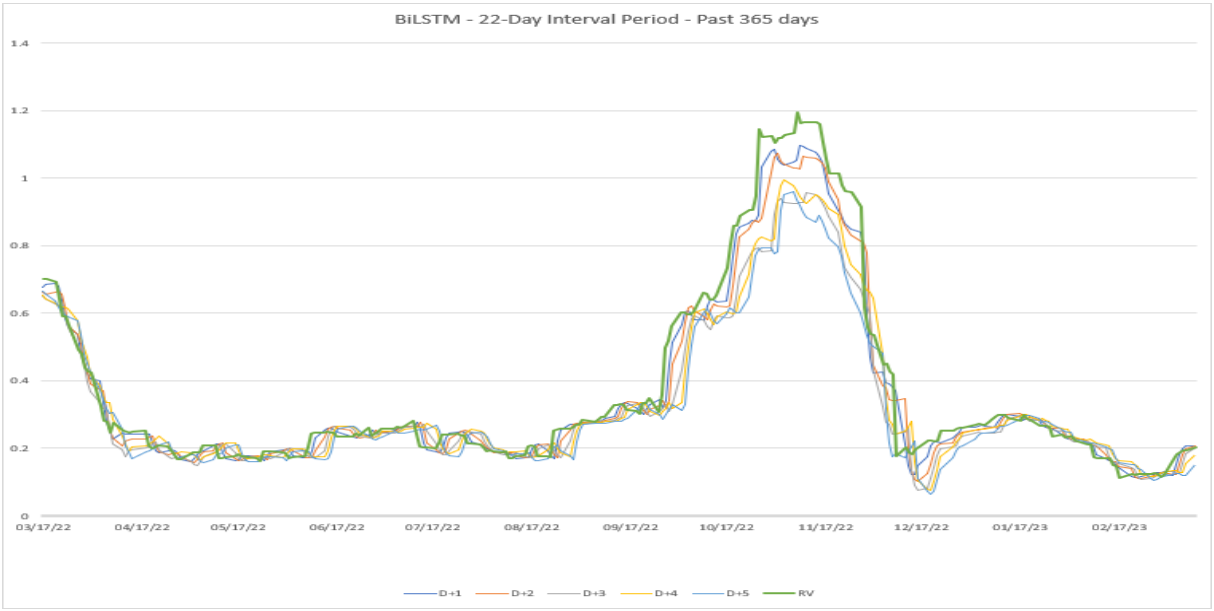

Figure 24: BiLSTM Forecasting Model Using 22-Day Interval Window

Similar to the univariate LSTM model, the BiLSTM fits the realized volatility quite well but at times underestimates it. This underestimation is particularly notable during the period of increased volatility, where each forecasting horizon lag reduces the estimation for the same time t. The underestimation appears to intensify significantly from the D+3 to the D+5 forecasting horizons. Upon comparing the visuals of the two LSTM models, the BiLSTM seems to be performing slightly worse due to these underestimations, which are not as severe in the forecasts of the univariate LSTM model.

BiLSTM (22-Day interval)		
Forecasting horizon	RMSPE	RMSE
D+1	0.096697	0.029952
D+2	0.130348	0.041313
D+3	0.158084	0.060878
D+4	0.181593	0.063616
D+5	0.197607	0.075702

Table 19: Forecasting performance for BiLSTM model (22-Day interval window)

As the graph suggests, the BiLSTM is only slightly outperformed by the univariate LSTM model, and it also trails behind the naive model for the D+1 forecast. However, for the reaming forecast horizons, the BiLSTM outperforms the naive model and even

surpasses the univariate LSTM model on the D+5 forecast with an RMSPE of 0.197607 against 0.199733.

Therefore, for the 22-Day interval window, the best-performing model measured both on the relative and absolute measure is the univariate LSTM for the D+1 forecast. Furthermore, for the first time, the D+1 forecast has not been the best-performing forecasting horizon for all the models. Both the GARCH models showed a better forecasting performance for the D+5 horizon. The results of the DM test for the LSTM models against the best-performing representatives of each model are as follows:

DM Test (22-Day inteval)				heightModel 1
Model 2	d - mean	P-Value	Result	
LSTM (D+1)	Naive (D+1)	-0.00003306	0.365044	Insignificant
LSTM (D+1)	GARCH (D+5)	-0.02671689	0.043116	Significant
LSTM (D+1)	GJR-GARCH (D+5)	-0.02284924	0.042640	Significant
LSTM (D+1)	BiLSTM (D+1)	-0.00025520	0.185793	Insignificant

Table 20: 5-Day interval window DM test results (5% significance level)

As seen in the table above, the univariate LSTM models fail to provide significantly different forecasts compared to the naive and BiLSTM model, as indicated by the p-value of the tests being above the significance level of 5%. However, the DM test between the LSTM and both GARCH models model shows p-values of 0.043116 and 0.042640 respectively. Therefore, the performance of the LSTM model compared to the two GARCH models is significantly different. This indicates that the LSTM model is outperforming the GARCH models as the RMSPE is also lower. This is noteworthy because it indicates that LSTM, a machine learning method, is providing superior forecasting performance relative to traditional econometric GARCH models in this specific context. Furthermore, the d - mean is indicating that all of the forecasting models produce higher forecasting errors compared to the LSTM model, with the two GARCH models being the most significant. It is also important to note that while the BiLSTM model did not significantly outperform the univariate LSTM model in the Diebold-Mariano test, it still demonstrated better performance compared to the naive model for forecasting horizons beyond D+1. This could potentially highlight the benefits of additional complexity in the model, such as the bidirectional nature of the BiLSTM, for longer forecasting horizons. However,

the insignificance of the DM test results between the LSTM and BiLSTM models suggests that the added complexity of the BiLSTM model does not significantly improve the forecasting performance for the 22-day interval window. This could indicate that simpler models might be sufficient when dealing with this specific dataset and forecasting horizon.

Through a detailed analysis across different forecasting horizons, the comparison of LSTM-based models with traditional GARCH models and a naive forecasting model led to several key findings. LSTM models, especially BiLSTM, consistently outperformed other models, underlining the effectiveness of machine learning in volatility forecasting. However, it is crucial to note that these performance differences were not always statistically significant, and the simplistic naive forecasting model held its ground surprisingly well. The thesis also highlighted the varying model performances across different forecasting horizons and the negligible impact of added complexity, as seen in the BiLSTM model, on predictive accuracy. Therefore, the choice of the model can significantly depend on the specific forecasting horizon, and simpler models may suffice in certain scenarios. In conclusion, while machine learning techniques like LSTM offer a promising direction in volatility forecasting, the comparative importance of simpler models and the adaptability across different forecasting horizons must not be overlooked. The balance between model complexity and predictive accuracy should be considered while choosing the optimal model for volatility forecasting.

7 Conclusion

In conclusion, this thesis compares the performance of econometric models and machine learning models in predicting volatility in gas spot data. The analysis, conducted using varying interval windows, provides valuable insights into the performance, robustness, and adaptability of these models.

Traditional econometric models, such as GARCH and GJR-GARCH, exhibit strengths and limitations in volatility prediction. Although they show incremental improvements with longer interval windows, they consistently fall short of matching or surpassing the prediction accuracy achieved by machine learning models. Both the standard GARCH and the GJR-GARCH continuously produced higher average forecasting errors than the machine learning models, indicating that these econometric models struggle to fully capture the complexities inherent in gas spot data, thus highlighting the need for more adaptive and nuanced modeling techniques. On the other hand, machine learning models, specifically LSTM and BiLSTM, emerged as the front runners in this comparative analysis. Across all interval windows, these models frequently delivered superior performance, often outpacing the GARCH models and the naive benchmark. The univariate LSTM model, in particular, demonstrated impressive consistency and robustness in its predictive accuracy.

Interestingly, the naive model, which assumes recent historical volatility as forward predictions, performed quite well. Especially in the 22-day window, the naive model demonstrated a level of prediction accuracy that set a challenging benchmark for the other, more sophisticated models. This emphasizes that the immediate past can bear heavily on volatility forecasting, a characteristic that all models should ideally account for.

A quite unexpected finding was the impact of data smoothing and stationarity that became evident as we changed the interval window for analysis. With an increase in the interval window, the data showed greater smoothness and less stationarity, which consequently improved the forecasting accuracy for all models. This indicates that interval selection can influence model performance and should be considered a key parameter in volatility prediction.

The evidence from the analysis clearly underscores the proficiency of machine learning models over traditional econometric models for forecasting volatility in gas spot data. The results emphasize the potential of machine learning techniques and highlight their adaptability in capturing the intricate patterns within gas spot data volatility. However, these conclusions are bound by the specific conditions of this thesis, the data used, and the selected interval windows. It would be prudent to conduct further research with a diverse range of datasets and varying time horizons to strengthen these findings and to delve deeper into the potential of machine learning in volatility forecasting. Furthermore, it could be insightful to investigate the development of models that deliver robust performance regardless of the interval window, integrating more volatility indicators and maintaining performance across all forecasting periods.

8 Bibliography

Altman, E. I., Marco, G., & Varetto, F. (1994). Corporate distress diagnosis: Comparisons using linear discriminant analysis and neural networks (the Italian experience). Journal of Banking & Finance, 18(3), 505-529.

Andersen, T. G., Bollerslev, T., Diebold, F. X., & Labys, P. (2003). Modeling and forecasting realized volatility. Econometrica, 71(2), 579-625.

Bengio, Y., LeCun, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

Bentley, A.E. (2018) Cold chaos highlights gas supply tightness, Argus Media. Retrieved from: https://www.argusmedia.com/en/blog/2018/february/28/cold-chaos-highlights-gas-supply-tightness

Bergmeir, C., Hyndman, R. J., & Koo, B. (2018). A note on the validity of cross-validation for evaluating autoregressive time series prediction. Computational Statistics & Data Analysis, 120, 70-83. https://doi.org/10.1016/j.csda.2017.11.003

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. Journal of Econometrics, 31(3), 307-327.

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. Journal of Political Economy, 81(3), 637-654.

Breusch, T. S. (1978). TESTING FOR AUTOCORRELATION IN DYNAMIC LINEAR MODELS*. Australian Economic Papers, 17(31), 334-355. https://doi.org/10.1111/j.1467-8454.1978.tb00635.x

Chatfield, C. (2005), Time-series forecasting. Significance, 2: 131-133.

Čeperić, E., Žiković, S., & Čeperić, V. (2017). Short-term forecasting of natural gas prices using machine learning and feature selection algorithms. Energy, 140, Part 1.

https://doi.org/10.1016/j.energy.2017.09.026.

Chen, Q., & Robert, C. Y. (2022). Multivariate realized volatility forecasting with graph neural network. In Proceedings of the Third ACM International Conference on AI in Finance (ICAIF '22), 156–164.

Cho, K., Van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the properties of neural machine translation: Encoder-decoder approaches. arXiv preprint arXiv:1409.1259.

Chong, E., Han, C., & Park, F. C. (2017). Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. Expert Systems with Applications, 83, 187-205.

Dickey, D. A., & Fuller, W. A. (1979). Distribution of the Estimators for Autoregressive Time Series With a Unit Root. Journal of the American Statistical Association, 74(366), 427–431.

Dunis, C. L., & Miao, J. (2006). Volatility filters for FX trading strategies. Journal of Asset Management, 7(2), 109-120.

Edwards, M. R. (2017). Energy trading and investing: Trading, risk management, and structuring deals in the energy market. McGraw-Hill Education.

Egholm, L. (2014). Philosophy of Science: Perspectives on Organisations and Society. København. Hans Reitzels forlag.

EIA (2021). Natural Gas Explained. U.S. Energy Information Administration. Retrieved from https://www.eia.gov/energyexplained/natural-gas/

Engle, R. F. (1982). "Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation". Econometrica. 50 (4): 987–1007.

ENTSOG (2020). European Gas Network Development Plan 2020. European Network of Transmission System Operators for Gas. Retrieved from https://www.entsog.eu/2020-ndp

Fałdziński, M., Fiszeder, P., & Orzeszko, W. (2020). Forecasting Volatility of Energy Commodities: Comparison of GARCH Models with Support Vector Regression. Energies, MDPI, 14(1), 1-18, December.

Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 270(2), 654-669.

Gao, X. (2017). Forecasting realized volatility: A Bayesian model-averaging approach. Journal of Applied Econometrics, 32(2), 435-457.

Geman, H. (2005). Commodities and Commodity Derivatives (1st ed.). Chichester: John Wiley & Sons Ltd.

Glosten, L.R., Jagannathan, R. and Runkle, D. (1993) On the Relation between the Expected Value and the Volatility of the Nominal Excess Return on Stocks. Journal of Finance, 48, 1779-1801. http://dx.doi.org/10.1111/j.1540-6261.1993.tb05128.x

Godfrey, L. (1978). Testing against general autoregressive and moving average error models when the regressors include lagged dependent variables. Econometrica, 46:1293-1302.

Harvey, D., Leybourne, S., and Newbold, P. (1997), Testing the equality of prediction mean squared errors. International Journal of forecasting, 13(2), 281-291.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hull, J., & White, A. (1987). The pricing of options on assets with stochastic volatilities. Journal of Finance, 42(2), 281-300.

Hyndman, R.J., & Athanasopoulos, G. (2021) Forecasting: principles and practice, 3rd edition, OTexts: Melbourne, Australia.

IEA (2019). Gas 2019: Analysis and Forecasts to 2024. International Energy Agency. Retrieved from https://www.iea.org/reports/gas-2019

Jacobs, B., & Li, J. (2021). Forecasting option-implied volatility using machine learning algorithms. Journal of Financial Data Science, 3(2), 9-35.

Jarque, C. M., & Bera, A. K. (1980). Efficient tests for normality, homoscedasticity and serial independence of regression residuals. Economics Letters, 6(3), 255-259. https://doi.org/10.1016/0165-1765(80)90024-5

Kaastra, I., & Boyd, M. (1996). Designing a neural network for forecasting financial and economic time series. Neurocomputing, 10(3), 215-236.

Kimoto, T., Asakawa, K., Yoda, M., & Takeoka, M. (1990). Stock market prediction system with modular neural networks. In International Joint Conference on Neural Networks (Vol. 1, pp. 1-6). IEEE.

Masters, D., & Luschi, C. (2018). Revisiting Small Batch Training for Deep Neural Networks. arXiv preprint arXiv:1804.07612.

Moghar, A., & Hamiche, M. (2020). Stock Market Prediction Using LSTM Recurrent Neural Network. Procedia Computer Science, 170, 1168-1173. https://doi.org/10.1016/j.procs.2020.03.049

Russell, S. J., & Norvig, P. (2016). Artificial Intelligence: A Modern Approach (3rd Edition). Pearson.

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. IEEE Trans. Signal Process., 45, 2673-2681.

Song, Y., Tang, X., Wang, H., & Ma, Z. (2023). Volatility forecasting for stock market incorporating macroeconomic variables based on GARCH-MIDAS and deep learning models. Journal of Forecasting, 42(1), 51– 59. https://doi.org/10.1002/for.2899

Tsay, R. S. (2010). Analysis of Financial Time Series (3rd ed.). John Wiley & Sons inc.

Yan, X. (2010). Linear Regression Analysis: Theory and Computing. World Scientific.

Zahid, M., Iqbal, F., & Koutmos, D. (2022). Forecasting Bitcoin Volatility Using Hybrid GARCH Models with Machine Learning. Risks 10, no. 12: 237. https://doi.org/10.3390/risks10120237

Zhang, G. P., & Zhang, Y. (2008). Neural network forecasting of the British Pound/US Dollar exchange rate. Omega, 36(5), 881-892.

Zhang, G. P. (2018). Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing, 50, 159-175.

Zhang, X., Trmal, J., Povey, D., & Khudanpur, S. (2015). Improving deep neural network acoustic models using generalized maxout networks. In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 215-219). IEEE.

9 Appendix 1

To establish a uniform comparative basis and streamline the thesis writing process, an equal number of lags are used across the various GARCH models. The optimal number of lags, determined by several factors, is addressed in this appendix.

For examining the autocorrelation among various lags and suggesting the probable count of lags that can explain the relationship between past lags of price and volatility with future levels of volatility, the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) have been generated for the entire dataset, which consists of 2868 days. As depicted in the ACF and PACF for the full dataset, significant autocorrelation extends until the fifth lag, and in some cases, it goes as far as the seventh lag.

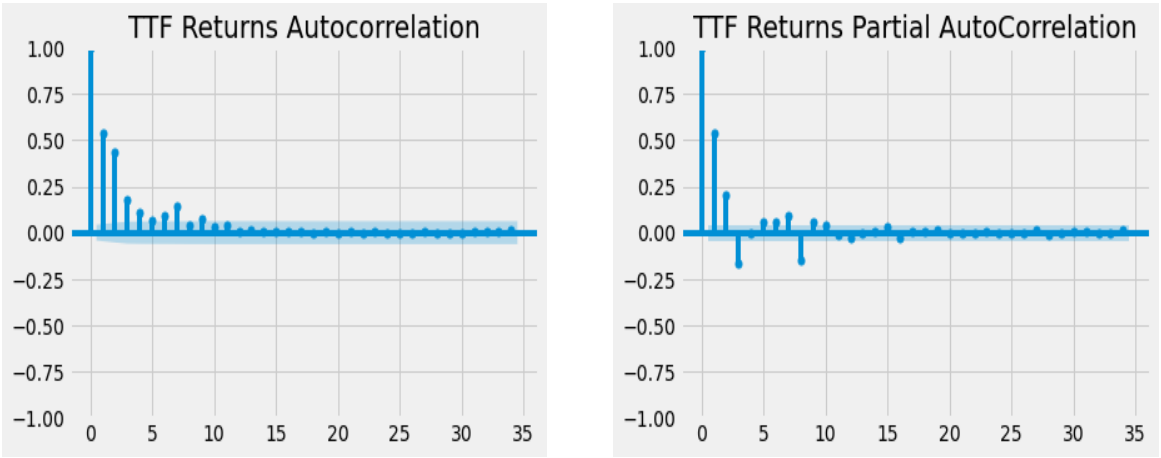

Figure 25: Autocorrelation and Partial Autocorrelation for the Entire Datset

The investigation begins with a model incorporating seven lags, i.e., examining the past seven days' price and volatility (GARCH(7,7)). To conduct the ARCH test, Python's **"arch"** package and its sub-package **"arch_model"** are employed. The resulting summary table is provided in Figure 26.

This model seems unsatisfactory for predicting future volatility as it fails to provide evidence of joint autocorrelation among significant parameters. The model is subsequently adjusted to five lags for both past prices and residuals (GARCH(5,5)) to inspect the outcome.

The results suggest that this model is insufficient for forecasting future volatility. Only one parameter is significant, meaning there is insufficient evidence that the parameters are jointly autocorrelated.

Constant Mean - GARCH Model Results

Dep. Variable:	returns	R-squared:	0.000
Mean Model:	Constant Mean	Adj. R-squared:	0.000
Vol Model:	GARCH	Log-Likelihood:	-5143.92
Distribution:	Normal	AIC:	10319.8
Method:	Maximum Likelihood	BIC:	10410.5

Mean Model

coef	std err	t	P>|t|	95.0% Conf. Int.
mu	-0.0270	5.471e-02	-0.493	0.622 [-0.134,8.022e-02]

Volatility Model

coef	std err	t	P>|t|	95.0% Conf. Int.
omega	1.4101	11.024	0.128	0.898 [-20.197, 23.17]
alpha[1]	0.2707	0.423	0.640	0.522 [-0.559, 1.100]
alpha[2]	0.0753	1.488	5.060e-02	0.960 [-2.841, 2.991]
alpha[3]	0.1429	0.211	0.677	0.499 [-0.271, 0.557]
alpha[4]	0.0982	1.143	8.595e-02	0.932 [-2.142, 2.338]
alpha[5]	0.2629	0.894	0.294	0.769 [-1.489, 2.015]
alpha[6]	0.0200	1.625	1.228e-02	0.990 [-3.165, 3.205]
alpha[7]	1.2840e-12	0.877	1.463e-12	1.000 [-1.720, 1.720]
beta[1]	0.0543	5.276	1.030e-02	0.992 [-10.287, 10.396]
beta[2]	3.7503e-13	0.682	5.496e-13	1.000 [-1.337, 1.337]
beta[3]	1.5308e-12	1.285	1.191e-12	1.000 [-2.519, 2.519]
beta[4]	1.1834e-12	0.192	6.165e-12	1.000 [-0.376, 0.376]
beta[5]	1.6311e-12	0.324	5.033e-12	1.000 [-0.635, 0.635]
beta[6]	3.5238e-13	0.897	3.927e-13	1.000 [-1.759, 1.759]
beta[7]	0.0757	1.526	4.960e-02	0.960 [-2.915, 3.066]

Figure 26: Summary Table of GARCH (7,7)

Constant Mean - GARCH Model Results

Dep. Variable:	returns	R-squared:	0.000
Mean Model:	Constant Mean	Adj. R-squared:	0.000
Vol Model:	GARCH	Log-Likelihood:	-5146.05
Distribution:	Normal	AIC:	10316.1
Method:	Maximum Likelihood	BIC:	10384.1

Mean Model

coef	std err	t	P>|t|	95.0% Conf. Int.
mu	-0.0265	6.283e-02	-0.422	0.673 [-0.150,9.665e-02]

Volatility Model

coef	std err	t	P>|t|	95.0% Conf. Int.
omega	1.4824	0.698	2.124	3.367e-02 [0.114, 2.850]
alpha[1]	0.2928	7.818e-02	3.745 1	.801e-04 [0.140, 0.446]
alpha[2]	0.0554	0.180	0.308	0.758 [-0.297, 0.408]
alpha[3]	0.1528	0.174	0.879	0.379 [-0.188, 0.493]
alpha[4]	0.0878	0.162	0.542	0.588 [-0.229, 0.405]
alpha[5]	0.2572	0.137	1.871	6.139e-02 [-1.228e-02, 0.527]
beta[1]	0.1074	0.596	0.180	0.857 [-1.060, 1.275]
beta[2]	0.0000	0.617	0.000	1.000 [-1.210, 1.210]
beta[3]	4.83e-13	0.362	1.335e-12	1.000 [-0.710, 0.710]
beta[4]	0.0220	0.210	0.105	0.917 [-0.389, 0.433]
beta[5]	0.0247	0.244	0.101	0.919 [-0.454, 0.503]

Figure 27: Summary Table of GARCH (5,5)

Subsequently, the model is adjusted to two lags for both past prices and past residuals (GARCH(2,2)) to inspect the outcome.

Constant Mean - GARCH Model Results

Dep. Variable:	returns	R-squared:	0.000
Mean Model:	Constant Mean	Adj. R-squared:	0.000
Vol Model:	GARCH	Log-Likelihood:	-5159.75
Distribution:	Normal	AIC:	10331.5
Method:	Maximum Likelihood	BIC:	10365.5

Mean Model

coef	std err	t	P>|t|	95.0% Conf. Int.
mu	-0.0303	5.153e-02	-0.588	0.557 [-0.131,7.070e-02]

Volatility Model

coef	std err	t	P>|t|	95.0% Conf. Int.
omega	0.7452	0.361	2.066	3.883e-02 [3.826e-02, 1.452]
alpha[1]	0.3534	0.102	3.460	5.396e-04 [0.153, 0.554]
alpha[2]	0.0969	7.519e-02	1.289	0.197 [-5.043e-02, 0.244]
beta[1]	0.0384	0.124	0.309	0.758 [-0.206, 0.282]
beta[2]	0.5113	6.764e-02	7.558	4.079e-14 [0.379, 0.644]

Figure 28: Summary Table of GARCH (2,2)

While all the alpha parameters, i.e., past lags of the price, indicate joint autocorrelation, the beta parameter (Beta 1), representing the first lag of the past residuals, is insignificant. Consequently, an attempt is made to reduce the lags to one for both past prices and residuals (GARCH(1,1)).

Constant Mean - GARCH Model Results

Dep. Variable:	returns	R-squared:	0.000
Mean Model:	Constant Mean	Adj. R-squared:	0.000
Vol Model:	GARCH	Log-Likelihood:	-5168.57
Distribution:	Normal	AIC:	10345.1
Method:	Maximum Likelihood	BIC:	10367.8

Mean Model

coef	std err	t	P>|t|	95.0% Conf. Int.
mu	-0.0210	5.713e-02	-0.367	0.714 [-0.133,9.102e-02]

Volatility Model

coef	std err	t	P>|t|	95.0% Conf. Int.
omega	0.4850	0.264	1.839	6.587e-02 [-3.182e-02, 1.002]
alpha[1]	0.2828	8.221e-02	3.441	5.806e-04 [0.122, 0.444]
beta[1]	0.7172	7.757e-02	9.245	2.359e-20 [0.565, 0.869]

Figure 29: Summary Table of GARCH (1,1)

At this stage, all the parameters are significant, suggesting that a GARCH(1,1) model is adequate for predicting the future volatility of TTF Spot prices. Hence, the GARCH model forecasts utilized in this thesis will be based on one lag of past prices and one lag

of past residuals. This is true for both the standard GARCH and the GJR-GARCH. All the pre estimation test, necessary for this model is made in section 5.