Forecasting Norwegian Fresh Salmon Export for 2024 using a Long Short-Term Memory Model

Jakob Lindstrøm
NTNU Business School
Norwegian University of Science and Technology
Trondheim, Norway
jakoblin@stud.ntnu.no

Marie Mariussen
NTNU Business School
Norwegian University of Science and Technology
Trondheim, Norway
marmari@stud.ntnu.no

***Abstract*—Exportation of fresh salmon represent an important component in the Norwegian economy. This paper leverage a deep learner method named long short-term memory to forecast the export volume of fresh Norwegian salmon for 2024 using four open source data sets. The model will be introduced as an alternative to traditional statistical models. The experimental output implies that our model produces satisfying results. However, there are still several drawbacks in the field of artificial learning compared to well-established statistical models.**

I. Introduction

The problem statement for this paper is as follows: *Forecast the export volume of Norwegian fresh salmon for the year 2024 using a Long Short-Term Memory machine learning model.*

Norwegian seafood is one of Norway's most desired global trade goods, where salmon contributes to 71% of the total national seafood export measured in NOK [15]. The export of salmon is as most commodities, based on demand and supply [9]. However, demand and supply is based on several underlying macro- and microeconomic factors. This paper will leverage domestic-related macroeconomic factors to forecast the exportation volume of Norwegian fresh salmon for the year 2024. The model used to forecast is a Long Short-Term Memory model, LSTM [13]. The LSTM is a machine learning model that belongs to the realm of deep learning.

The intention of this paper is to further introduce the use of deep learning to generate macroeconomic estimates. Many of the current estimation-techniques for macroeconomic estimates used by private and governmental agencies are based on traditional statistical models. As the development of new techniques within the field of machine learning is rapidly changing, the traditional statistical models should be challenged by machine learning models and especially by deep learners.

This paper describes the approach from a technical point of view and goes into detail about the mathematics and underlying workings of leveraging raw data to produce forecasts. The type of data used are time series, which means that the observations occur sequentially. The task of predicting time series requires the use of models that are specifically meant for such modelling. Recurrent neural networks, RNN, are deep learning models that are used to analyse and model time series. There are several subtypes of RNNs, one of the most prominent is the LSTM model. The LSTM will be used to forecast the data, and its performance will be compared against an ensemble model specifically built to predict time series.

The initial part of this paper will cover related work. Next, the material and methods used for this project will be presented, including the data sets, preprocessing, feature engineering and modelling. Furthermore, we will present the results using multiple evaluation metrics and discuss the results in light of our approach and pipeline. Lastly we will present a conclusion.

The corresponding GitHub repository used to generate results can be found at this link: https://github.com/DataJakob/DT8807-Advanced-Topics-in-Deep-Learning.

II. Related Work

Time series prediction separates itself from prediction of cross sectional data, in the way that the data is of sequential nature. Time series contains several components: trend, seasonality, cycles and residuals [2]. The importance of leveraging these components are crucial for predicting time series with satisfactory results [6].

Time series forecasting can be divided into three categories; short-term forecasting, mid-term forecasting and long-term forecasting. The time scales for these categories ranges from one hour to one week, one month to one year and more than a year, respectively [19]. Work done by several researchers, implies that short term forecasting achieves higher accuracy than other types [19] [24].

In later years, deep learning has emerged as an effective tool for time series forecasting [11]. Human brains processes information incrementally while maintaining an internal model of the data it´s processing, constantly updating with new information [8]. The Recurrent Neural Network adopts some of the same principles, and is considered quite useful for time series analysis due to its ability to model temporal dependencies in the data and, its way of back propagating over time. However, vanilla RNNs are prone to the gradient vanishing or exploding during the training process [4]. In order to compensate for the shortcomings of vanilla RNNs, Hochreiter and Schmidhuber introduced the Long Short-Term Memory neural network [13]. The purposed LSTM method has been recognised as a good solution due to the added gate layers and memory cell [24].

In addition to the long short term memory model, there are several other machine learning methods that can predict time series. A common approach is by using multi model ensembles, further referenced to as MME [1]. Such models utilise several machine learning models to generate a final prediction. Each separate model is defined as a weak learner, and the gathered performance of each weak learner creates one strong learner. For time series modelling, MME usually works in a sequential manner where there are four weak learner, that attempts to predict their respective time series component.

For the task at hand concerning forecasting the volume of fresh Norwegian salmon, a very similar work have been published. In 2023 Mikaella Zitti published her work on using a LSTM model for predicting the export price for Norwegian salmon [26]. She used a vector autoregression model (VAR) integrated with a LSTM, and compared it against traditional statistical models. Her findings implied that the VAR-LSTM combination generated satisfying results, however the LSTM model was generally outperformed by standard statistical methods. This paper can be viewed as an additional contribution to raise awareness regarding the utilisation of deep learners in prediction of macroeconomic estimates.

III. Materials and Methods

In order to forecast the Norwegian fresh salmon export for 2024, data from various sources have been gathered and fitted into a machine learning pipeline. Before the data was fitted through the pipeline, it has been preprocessed. The techniques applied are a direct result based on the problem statement and a comprehensive exploratory data analysis. This section will provide the technical details concerning these techniques, while the discussion-section will explain the reasoning for applying these techniques.

A. Data

Four sets of data have been collected. The target variable which is exported Norwegian fresh salmon has been retrieved from SSB [21]. The choice of data was determined by the use of marketing analysis method PESTEL. This method detects important factors for a specificed object. The factors are political, economical, social, technological, enviromental and legal factors [25]. The three original explanatory variables have all been retrieved from different sources. The meat price index has been gathered from the Food and Agriculture Organisation ruled by the UN [10], the data is collected on monthly basis. The data regarding the foreign currency rate between EURO and NOK has been retrieved from yahoo finance, and is based on a weekly observational interval [23]. Lastly, the Norwegian policy rate variable has been retrieved from Norwegian Bank Investment Management [17]. The interest rate is observed on a quarterly basis.

B. Data Preprocessing

By gathering data from various sources and with varying observational intervals, several preprocessing steps have carried out. The first task was to convert all the data into readable file formats, then removing redundant information from the files. Thereafter, the main task was to convert all the data to have a common observation interval. As previously stated, some observation was recorded weekly, other monthly and some quarterly. For the observations with intervals that were larger than weekly, the observation was repeated in order to fit into a weekly interval. When the observations aligned, all the data was concatenated and was thereby ready for feature engineering.

C. Feature Engineering

Feature engineering have been used substantially in order to feed the machine learning model with informative variables, with the purpose of explaining the variance in the target variable. The first technique that has been utilised is the generation of a categorical time variable. This variable has a perfect linear relationship with time, and has a start value of zero. Secondly, a set of seasonal dummy indicators have been generated. These dummies are binary and indicates whether an observation lies within a certain week, quarter or half year, or not. Next, a log transformation function has been added to the pipeline. The function log transforms specified columns.

$$X_L = ln(X) \tag{1}$$

Where ln is the function for natural logarithms. The second to last feature engineering technique is standardising the variables. This subtracts all the variables with their respective mean and divide them by the variable's respective standard deviation.

$$S = \frac{X - \mu}{\sigma} \tag{2}$$

S is the standarised value, X is the variable in its original form, μ is the mean value and σ is the volatility. The last technique applied was reshaping the dataset by adding lagged variables. The function transforms the two dimensional data into three dimensions, where each new dimension contains the n_{th} lagged version of the original data set.

D. Modelling

Utilising the data set, one can feed the data into a machine learning model in order to forecast future export volume.

1) LSTM: Our project is based on a Long Short-Term Memory model. The main difference between RNN and LSTM lies in the ability to handle long term sequential data. Unlike RNN, which utilise the same feedback loop for events occurring a long time ago and recently, LSTM employs different pathways for detaining long term and short term memories. The structure of LSTM reduces the probability of the gradient vanishing or exploding during training, and can be illustrated as bellow:

LSTM is structured with four components, including the memory cell, the forget gate, the input gate and the output gate [13]. The memory cell affects the entire LSTM unit and is the key element regarding the cell state.

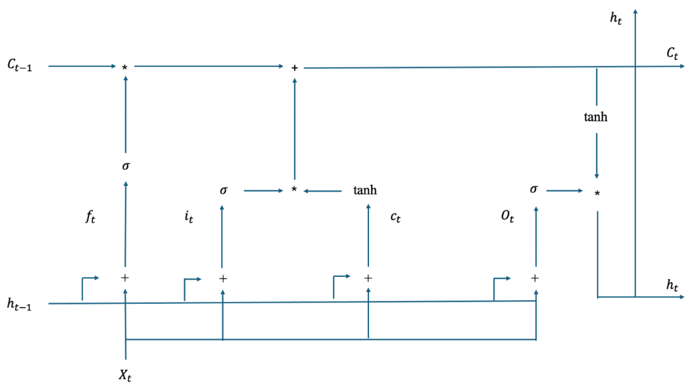

Fig. 1: Structure of LSTM

The forget gate determines what percentage of the cell state is remembered, and can be expressed as follows:

$$f_t = \sigma(u_f * h_{t-1} + w_f * X_t + b_f) \tag{3}$$

Where σ represents the sigmoid activation function, u and w represents the weights of the hidden state value and the input, X_t represents the vector of inputs and lastly b_f refers to the bias values of the gate [8].

Furthermore, the input gate determines what additional information to add to the cell state:

$$i_t = \sigma(u_i * h_{t-1} + w_i * X_t + b_i) \tag{4}$$

Alongside the input gate layer, a candidate memory cell can be expressed as below:

$$c_t = \tanh(u_c * h_{t-1} + w_c * X_t + b_c) \tag{5}$$

An edited representation of the new current cell state is:

$$C_t = f_t * C_{t-1} + i_t * c_t \tag{6}$$

Lastly, the output gate layer represents the output of the cell state values:

$$O_t = \sigma(u_O * h_{t-1} + w_O * X_t + b_O) \tag{7}$$

The final output of the hidden state consists of the output of the cell state that provides relevant information to the next layer in the LSTM model:

$$h_t = O_t * \tanh(C_t) \tag{8}$$

The structure of the network shows how LSTM can reserve information for longer periods, and is likely the reason why LSTM performs better than RNN. The cell state is not directly influenced by the weight and biases, which allows the long term memory to flow through the time series without causing the gradient to vanish or explode.

The LSTM layer of the network constitutes only a part of the network, and needs complimentary parts in order to generate predictions. The other parts consist of various layers. These are dense layers with n_{th} units and activation functions. Additional layers that can improve model performance are batch normalisation and dropout layers. Their corresponding purposes are normalisation of variables and randomly deactivating $i\%$ of the neurons in the model during training. The activation functions used in the dense layers are relu, gelu, selu and silu. [14].

$$relu = max(x, 0) \tag{9}$$

$$selu = \text{if } x > 0 : (S * x)|s * \alpha * [exp(x) - 1] \tag{10}$$

$$silu = x * \sigma(x) \tag{11}$$

$$gelu = x * P(X <= x) \tag{12}$$

S and α are constants, σ is the sigmoid function and $P(X)$ stands for the normal probability density function.

2) RMSprop: The Root Mean Square Propagation algorithm (RMSprop) was used to update the weights of our LSTM model. This algorithm deploys a adaptive learning rate. Unlike the traditional Stochastic Gradient Decent (SGD), which updates parameters based solely on the previous value of a gradient, the RMSprop algorithm includes a squared gradient term. [12]. The difference is shown in the formula for RMSprop below:

$$W = W - \alpha(dw/\sqrt{S_{dw} + \epsilon}) \tag{13}$$

$$B = B - \alpha(db/\sqrt{S_{db} + \epsilon}) \tag{14}$$

Where $S_{dw} = \beta * S_{dwprev} + (1 - \beta)(dw)^2$
and $S_{db} = \beta * S_{dbprev} + (1 - \beta)(db)^2$

3) Hyperparameter tuning: The process of setting hyperparameters was mainly driven by a tuning-algorithm and qualitative evaluations. The hyperparameters for the network architecture was set by deploying a random search algorithm [3]. Before deploying the algorithm, the search method needs to be fitted with hyperparameters. This includes values for weight iterations, which are epochs and batch sizes. In addition, a callback function can be deployed to the search. This function will return the best set of weights if the optimiser ends up converging or receiving lower results on it's objective.

After deciding hyperparameters for the algorithm, the algorithm produces the best arbitrary hyperparameters for the network architecture. The best arbitrary values is equivalent to a well-suited model.

E. Multi Model Ensemble

In order to establish a benchmark for comparative purposes, we created an ensemble model aimed for time series predictions. The model is referred to as Multi Model Ensemble, further referenced as MME. This model includes the use of Multiple Linear Regression, K-Nearest-Neighbour Regression and two Random Forest Regression models. The MME works in a sequential manner, where each model is trained on different sets of the training data, and the additive sum of the predictions acts as the final prediction.

The formula for Multiple Linear Regression, MLR, can be expressed as below: [20]:

$$Y = \beta_0 + \beta_1 * X_1 + \beta_2 * X_1^2 + \varepsilon \tag{15}$$

where ϵ is a independent random variable, with $E(\epsilon) = 0$ and $\text{var}(\epsilon) = \sigma^2$ [22]. The MLR is trained on time data and predicts a trend.

The K-Nearest-Neighbour Regressor (KNN) is a data-driven machine learning model [20]. The algorithm locates the n_{th} closest other observations in the hyperspace, and takes the average of their respective target variable in order to generate a prediction. Location of close observations are calculated using Euclidean distance:

$$E_D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{16}$$

Where E_D is the distance. x and y represent different variables and the subscripts represent the observations [20]. This model is trained solely on seasonal data, and its target variable is the residuals of the original training data subtracted from the trend predicted by the MLR.

The Random Forrest Regressor (RFR) is an ensemble model that creates multiple decision trees trained on bagged data [20]. Each tree is trained and the average prediction of the model equals the final prediction of model. The algorithm can be fine tuned to achieve better performance, for example by the use of cross validation and grid search [7].

The RFR method is utilised twice in the MME. The first RFR model is trained solely on lagged versions of the explanatory variables, in order to detect cycles in the data. The target data incorporated in this model consist of the residuals from the KNN predictions. The second RFR model is trained on all the explanatory variables, and aims to predict the original target data subtracted with the three previous predictions.

F. Generating Future Explanatory Variables

In order to forecast the export volume of fresh Norwegian salmon one needs data. Simulated variables have been used due to data leakage. Data leakage implies that the explanatory variables are not available for modelling at the current time. Therefore, two of the variables have been simulated using various stochastic calculus operations incorporated with Monte Carlo simulations.

1) Wiener Process: A modified Wiener process have been used to simulate data for the forecast dataset [18]. The modification has been set in order to construct a heteroscedastic process.

$$X_{t+1} = Z * (\frac{S_t^2}{8} + 0.3) * \sigma * X_t + X_t \tag{17}$$

Where Z represents a random draw from the normal distribution, S_t ranges from -4 to 4 with a step size of 0.154. σ is the volatility of the time series.

2) Ornstein-Uhlenbeck Process: In addition to the Wiener process, a Ornstein-Uhlenbeck process has been utilised in order to generate data for the forecast [18].

$$X_{t+1} = r(\mu - X_t) + \sigma * X_t * Z + X_t \tag{18}$$

The same notations from the previous equation is applicable to this equation, with the addition of r as the mean reverting rate and μ as the mean reverting level.

G. Evaluation metrics

Model evaluation is an important aspect of time series forecasting due to its role of determining effectiveness and reliability of the model. This effectiveness can be measured with a variety of evaluation metrics, including Mean Error (ME), Mean Absolute Error (MAE), Mean Squared Error (MSE) and an modified confidence interval (CI). The formula for the evaluation metrics can be expressed as below:

$$\text{ME} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i) \tag{19}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{20}$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{21}$$

Where n represents the total number of observations, y_i represent actual values and $\hat{y_i}$ represent predicted values [20].

ME gives an indication whether the model on average predicts higher or lower values than the actual targets. MAE can be interpreted as the mean error of the predictions in either direction. MSE is the mean squared error and penalises the performance when there is a large residual between the predictions and actual values.

The last metric to be used in order to evaluate model performance is confidence interval. The interval is generated using Monte Carlo simulations where the confidence interval is the upper and lower percentile of the predictions plus a cumulative sum of random errors.

$$CI_{t_{H/L}} = MC_n[p_t + Z\sigma + \sum_{t=0}^{t}(Z\sigma)]_{H_{th}|L_{th}} \tag{22}$$

Time is represented as t, where H and L is the upper and lower percentile of an array. MC stands for Monte Carlo with n simulations. Z is a random draw from the normal distribution, and σ is the volatility of target residuals.

IV. Results

For evaluation purposes, we have compared the performance of our LSTM model with the constructed MME. The results are illustrated in the table below:

Metric	LSTM	Multi Model
ME val	443	-1'769
ME test	114	-2'616
MAE val	1'617	2'475
MAE test	2'627	3'366
MSE val	4'271'122	9'114'774
MSE test	9'949'735	18'464'217
CI val	[3,027-2141]	[3574,-2689]

TABLE I: Results comparing performance of LSTM vs. MM

Table 1 reflects the results of our models based on the metrics ME, MAE, MSE and CI. The metrics are calculated for both the validation set and the test set.

Initially we can observe an ME reflecting positive numbers for the LSTM model and negative numbers for the MME model. These numbers indicate that the LSTM model generally predict higher values than the actual values of the data set, while the MME model generally predict lower values than the actual values. The magnitude of LSTM model performance in terms of ME is also significantly lower compared to the MME. Furthermore, the MAE indicates that the LSTM model is better at predicting the target variable. The LSTM model has an MAE of 1617 on the validation set, implying that on average, our predicted export volume is off by 1617 tons of salmon per week compared to the actual export volume. In comparison, the MME model misses with an average error of 2475 tons of salmon per week on the validation set. We can also observe a significant leap in the evaluation of MSE for the models. The large MSE value for the MME model indicates a poorer performance in predicting outliers compared to the LSTM.

The confidence intervals are computed based on the validation set, as a two sided confidence interval and reflects an significance of 80%. One can observe that the LSTM model presents a lower confidence interval than the MME model, which is favourable. A lower confidence interval indicates a more accurate and reliable validation estimate.

The bias values for MAE is calculated as the difference in results between the validation and test set. The bias values from the LSTM model reflects a value of 1010, while the MME reflects a value of 891. These values indicateS an notable difference, where the LSTM is more susceptible to overfitting.

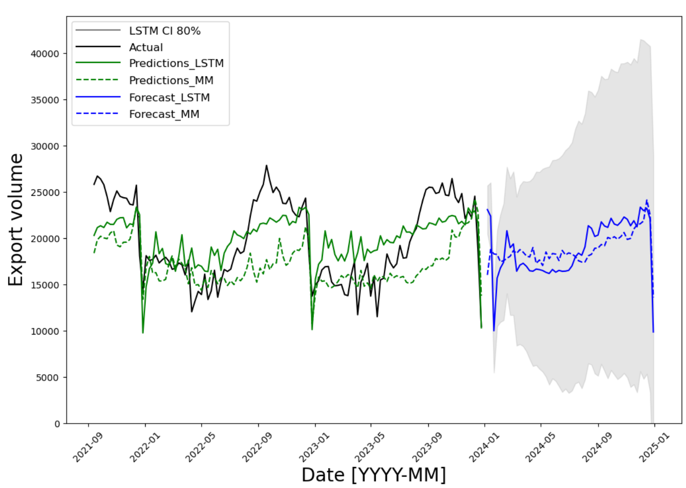

Fig. 2: Plot visualising the forecasted volume of fresh Norwegian salmon export

Our forecasted export volume for the upcoming year can be viewed in figure 2. The solid line represent our LSTM model, while the dashed line represent the MME model. The lines are further color-coded, where green represent the predictions on the test set, black represent the actual values and blue represent the forecasted values. The CI for the LSTM model is visualised as the grey area.

From the plot one can observe that the LSTM generally predicts higher estimates compared to the MME for the test data. However, in the case of forecasts, the same description is not applicable. The forecasts of the LSTM is no longer higher than the MME, and the differences in the forecasts has to some degree vanished.

Overall, the evaluation of the results indicate that the LSTM model performs better than the MME, in terms of used metrics.

V. Discussion

In the process of solving the problem statement for this paper, several decisions concerning the approach for a solution have been taken. This section will discuss the reasoning for the approach.

A. Problem statement

Forecasting the export volume for the upcoming year provides two main challenges. The first challenge concerns the handling of sequential data. Secondly, the forecasting technique demands the use of explanatory data, which is non-existent. In order to address the problem concerning the sequential data, a LSTM model was introduced. Secondly, stochastic processes were leveraged in order to simulate plausible values for the explanatory variables.

The choice of explanatory variables were determined based on a PESTEL analysis. Leveraging this analysis framework, one found variables that's plausible to have an impact on the variance of the target variable. The variables are of macroeconomic nature and provide some explanation for the target variance.

B. Pipeline

The target variable has been observed in a weekly interval. However, the Norwegian policy rate and food price index has a less frequent observation interval. This has led the observations to be expanded and repeated to match the same frequency as target variable. This configuration leads to less variance, and therefore creating less-informative variables. The choice of utilising such expanded variables, and not replacing, might lead to lower explanatory power of the model. Utilisation of the preprocessed data enables the process of running the data through a machine learning pipeline. The applied pipeline is illustrated in the figure below:

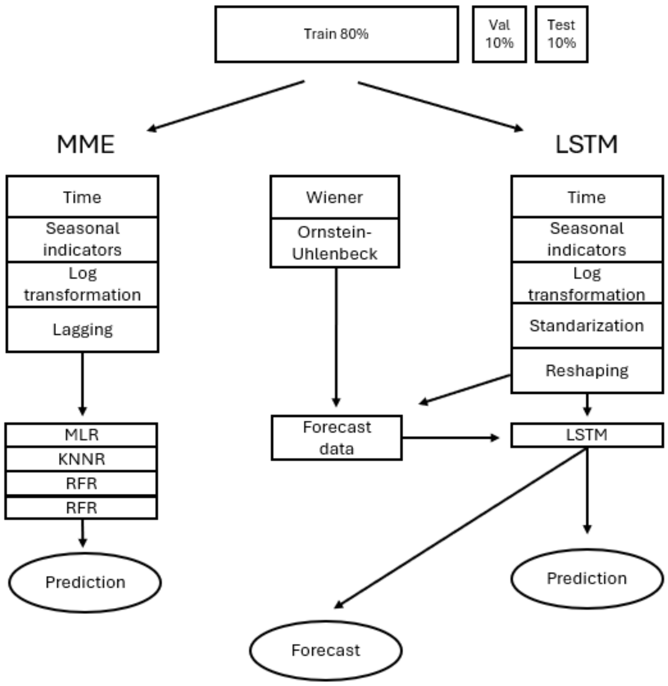

Fig. 3: Machine learning pipeline

The four variables and the timespan of the observations have been the building blocks for generating new variables that contain valuable information in regard to the target. The process of feature engineering contains five steps. The two first steps leverage the timespan and have the intention of generating variables which will help the model learn trend and seasonality. The target has an obvious upwards trend. Therefore, a discrete time column linear with the timespan has been generated, and its corresponding squared version. The exploratory data analysis indicates that there are three seasonalities which are especially prominent: annual, semiannual, and quarterly. For capturing these trends, dummy variables have been deployed, to indicate whether an observation lies within the current seasonal interval. This has resulted in the generation of 55 new variables, one for each week, quarter and half year, subtracted with one variable for each season to avoid the dummy variable trap [22].

The next step is log-transformation of the foreign exchange rate. This variable seems to have a lot of characteristics with the lognormal distribution, therefore it is of interest to transform the variable to be normally distributed. The two last steps prepares the data to be feed into the neural network. All the explanatory variables have been standardised to give the loss functions suitable values, that will enhance model performance. Lastly the data was reshaped from a 2d to a 3d format, where each new layer in the third dimension contains the n_{th} lagged version of the data. In total there are 13 layers. This decision was taken based on the autocorrelation detected in the exploratory data analysis.

A very important note concerning the explanatory variable is that lagged versions of the target variable (LTV) are not a part of the explanatory ones. Experimenting with the inclusion of the LTV resulted in poorer model performance. A possible reason for this result is that the variables are too informative and overruns the explanatory impact of the other features.

Predicting the data through LSTM requires a lot of decision-making regarding network architecture. To avoid completely arbitrary hyperparameters, a random search algorithm was deployed. This algorithm searched through many sets of random choices for perceptron amount, activation functions, learning rates and optimisers. Among the search results, the algorithm concluded that the RMSprop optimiser was ideal for this task. Leveraging this tuner, the performance of the model increased and the hyperparameters were set to deliver satisfying results.

The actual explanatory data used to forecast the exports for 2024, did not exist beforehand. Therefore, several stochastic processes have been leveraged to simulate plausible data used for the forecast. From the exploratory data analysis, it seemed like the food price index follows an annual process that could be simulated using a Ornstein-Uhlenbeck process. In the case of the currency variable, the same mindset could be used. It seems like the currency variable has neither an increase or decrease in value, but has a larger variance in the start and the end of the year. In other words, the process is heteroscedastic. Therefore, a Wiener process with a integrated exponential function was deemed appropriate to simulate the currency variable. For the interest rate, forecasts retrieved from Norwegian Bank Investment Management was used [16]. The other variables were of non-stochastic nature, and could therefore be easily generated.

C. Result and comparison

The MME was introduced for comparative purposes, to serve as a solid benchmark for the more advanced LSTM model. Unlike a statistical model such as SARIMAX, the MME utilise machine learning methods. Consequently, the MME provides a better baseline for comparison, as both the selected models are based on machine learning.

The MME is composed of four models, addressing different areas. From the exploratory data analysis (EDA) one can observe that the target seems to have a trend of quadratic nature. Therefore, the MLR is set to predict this particular trend. The KNN is used to capture seasonal variations, based on the seasonal indicators for weekly, quarterly and semiannual data. The first RFR model uses lagged independent variables to model cycles in the data based on autocorrelation values detected in the EDA, and the second RFR model is utilised to capture other residuals.

Results from table 1 reflect the effectiveness of both the LSTM model and the MME. As we previously stated, we can observe that our LSTM model performs better than the MME in terms of evaluation metrics. A possible reason for the better performance may be due to the complexity of the LSTM structure. Accordingly, LSTM has a stronger ability to detect complex sequential patterns in the data. In contrast, the MME composition might make the model suffer in terms of detecting the same patterns as the LSTM. Another possible reason for the lower performance of the MME is linked to the MLR. The MME shows pessimistic predictions compared to the actual values and compared to the LSTM. The main contributor to the MME in terms of predictive power is the MLR. The MLR is trained for a quadratic trend. However, the quadratic influence of the trend occurred late in the training set. Therefore, the quadratic impact of the MLR will be larger further away in time and this can be a reason for the pessimistic predictions. On the other hand, the LSTM shows larger values for bias than the MME, which indicates that the LSTM is more pruned to overfit.

From the observation regarding the difference in prediction magnitude between the LSTM and MME on the forecast set one can draw a possible explanation for this occurrence. The LSTM has in general higher predictions for the test data compared to the MME. This changes for the forecast data. It is possible that the forecast data has a combination of data distribution that differs from previous data used to train and evaluate the model. This implication implies that the fit of the model might be wrong for the new data, and vice versa. Therefore, the explanatory data used for the forecast is somewhat miss-specified, and other simulation or forecast techniques should be considered. For the problem statement this does not have a significant impact. The problem statement is to introduce deep learning as a tool for generating macroeconomic estimates. Utilization of other estimation techniques for the explanatory variables for the forecast data could probably result in a scenario without magnitude abnormalities in the forecasts. However, actually explaining the specific reasons for the prediciton error are difficult to address due to lack of interpretability of LSTM model, also known as the black box phenomena [5].

D. Summary and Future Work

In summary the deep learner LSTM provides forecasts with generally good estimates, and better than the MME. Experimenting with explanatory variables, it seems that the chosen variables from the PESTEL analysis provides significant predictive power to both of the models.

The LSTM model was trained on data with a weekly observational interval, creating a short-term forecasting model. Similar to what previous researchers have found, we also observed a high accuracy when using a short-term model. Another possible approach could be to collect more data, and utilise data on a daily basis instead of weekly. This approach might generate even more accurate results.

The MME was introduced as a benchmark for evaluating the LSTM model. Based on results from table 1 and figure 2, we can conclude that the LSTM model confirms reasonable results in terms of prediction, and a higher performance than the MME. We can mainly observe two reasons why the LSTM model performs better than the MME. The most important factor lies in the complex structure of the LSTM, making it more suitable to detect sequential data pattern. The other reason is due to the late influence of the quadratic trend in the training set of the MLP in the MME. On the other hand, the LSTM also have some flaws. The LSTM show larger bias values than the MME, indicating that the model is more pruned to overfit. In addition, the possible misspecification of the estimation techniques for the forecast data may result in unrealistic forecasts. The article of Zitti comes to a similar conclusion. Her VAR-LSTM model performs well, but the weaknesses of her solution also concerns interpretability.

Future improvements of the model performance can be to generate a better suited model architecture, re-specify the forecast data and possibly choose other explanatory variables.

E. Real World Applications

So far, this paper has shown how a Long Short-Term Memory model can be utilised in order to generate forecasts for macroeconomic variables. The model delivers satisfying results and gives an example on the use of deep learners and their performance. With further experimental research on the applications of LSTMs on macroeconomic variables, one can probably generate even more accurate results. Having accurate macroeconomic estimates will enable decision makes to have a better information basis to build their decisions upon.

The two main pain points of utilising deep learners in regards of governmental decision making are related to the advanced architectures of the models, and thereby it's interpretability. Since decision in governmental agencies impacts large amounts of people, the data which decision are made upon must be rock solid.

VI. Conclusion

In this paper, we have provided a solution for forecasting the export volume of fresh Norwegian salmon in 2024, using a LSTM model. Through a comprehensive analysis, this paper has gone in depth concerning the technical details from leveraging raw data to generate forecasts. The experimental results showed that our model performed with a satisfying result, and with a notable better performance than our benchmark model in terms of evaluation metrics.

As a result, we can see that deep learners are well suited to generate forecasts of time series, and can challenge many of the traditional statistical models. The two largest drawback of the model, is it's advanced setup and interpretability.

References

[1] Ahmed, K. et al. (2020) Multi-model ensemble predictions of precipitation and temperature using machine learning algorithms, *Atmospheric Research*, 236. Available at: https://www.sciencedirect.com/science/article/pii/S0169809519309858 (Accessed: 09.05.2024)

[2] Batista, G.E.A.P.A.; Parmezan, A.R.S. and Souze, V.M.A. (2019) Evaluation of statistical and machine learning models for time series prediction: Identifying the best stat-of-the-art and the best conditions for the use of each model, *Information Sciences*, 484, pp. 302-337. Available at: https://linkinghub.elsevier.com/retrieve/pii/S0020025519300945 (Accessed: 09.05.2024)

[3] Bengio, Y. and Bergstra, J. (2012) Random Search for Hyper-Parameter Optimization, *Jounral of Machine Learning Research*, 13, pp. 281-305. Available at: https://www.jmlr.org/papers/volume13/bergstra12a/bergstra12a.pdf?ref=broutonlab.com (Accessed: 09.05.2024)

[4] Bengio, Y.; Mikolov, T. and Pascanu, R. (2013) On the difficulty of training recurrent neural networks, *Proceedings of the 30th International Conference on Machine Learning*, 28(3), pp. 1310-1318. Available at: https://proceedings.mlr.press/v28/pascanu13.html (Accessed: 09.05.2024)

[5] Benitez, J.M.; Castro, J.L. and Requena, I. (1997) Are artiffical neural networks black boxes? *IEEE Transactions on Neural Networks*, 8(5), pp. 1156-1164. Available at: https://ieeexplore.ieee.org/abstract/document/623216 (Accessed: 10.05.2024)

[6] Beveridge, S. and Nelson, C.R (1981), A new approach to decomposition of economic time series into permanent and transitory components with particular attention to measurement of the 'business cycle', *Journal of Monetary Economics*, 7(2), pp. 151-174. Available at: https://www.sciencedirect.com/science/article/pii/0304393281900404 (Accessed: 09.05.2024)

[7] Budiman, F. (2019) SVM-RBF Parameters Testing Optimization Using Cross Validation and Grid Search to Improve Multiclass Classification, *Scientific Visualization*, 11(1), pp. 80-91. Available at: https://sv-journal.org/2019-1/07/en.pdf (Accessed: 09.05.2024)

[8] Chollet, F. (2021) *Deep Learning with Python*. 2nd ed. Shelter Island: Manning Publication Co.

[9] Douma S. and Schreuder H. (2017), *Economic Approaches to Organizations*, 6th. Pearson Education Limited.

[10] Food and Agriculture Organization of the United Nations (2024) *Markets and Trade* Available at: https://www.fao.org/markets-and-trade/commodities/meat/fao-meat-price-index/en/ (Accessed: 15.02.2024)

[11] Han, Z.; Zhao, J.; Leung, H.; Ma, K.F. and Wang, W. (2021), A Review of Deep Learning Models for Time Series Prediction, *IEEE Sensors Journal*, 21(6), pp. 7833-7848, https://ieeexplore.ieee.org/abstract/document/8742529 (Accessed: 09.05.2024)

[12] Hinton, G.; Srivastava, N. and Swersky, K. (2012). Lecture 6a overview of mini-batch gradient descent, *Neural networks for machine learning* Available at: https://www.cs.toronto.edu/˜hinton/coursera/lecture6/lec6.pdf (Accessed: 10.05.2024)

[13] Hochreiter, S. and Schmidhuber, J. (1997) Long Short-Term Memory, *Neural Computation*, 9(8), pp. 1735-1780.

[14] Keras (2024) *Layer activation functions*. Available at: https://keras.io/api/layers/activations/ (Accessed: 09.05.2024)

[15] Kyst (2024) *Eksportrekord for norsk sjømat i 2023*. Available at: https://www.kyst.no/eksportverdi-norges-sjomatrad-sjomateksport/eksportrekord-for-norsk-sjomat-i-2023/1609066 (Accessed: 02.05.2024)

[16] Norges Bank (2024) *Pengepolitiske Vurderinger*. Available at: https://www.norges-bank.no/aktuelt/nyheter-og-hendelser/Publikasjoner/Pengepolitisk-rapport-med-vurdering-av-finansiell-stabilitet/2024/ppr-12024/nettrapport-ppr-12024/ (Accessed: 27.04.2024)

[17] Norges Bank (2024) *Styringsrenten*. Available at: https://app.norges-bank.no/query/?fbclid=IwAR2WdTEeWVVvdNciJsTJGlnCgfIpxbqVEgmREKxj65FDLBbyWUEN0WxippQ#/no/interest?interesttype=KPRA&unit ofmeasure=R&duration=SD&frequency=M&startdate=2000-01-01&stopdate=2023-12-31 (Accessed: 15.02.2024)

[18] McDonald, R. (2013) *Derivatives Markets* Pearson Education Limited.

[19] Raza, M.Q. and Khosravi, A. (2015) A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. *Renewable and Sustainable Energy Reviews*, 50, pp. 1352–1372. Available at: https://doi.org/10.1016/j.rser.2015.04.065 (Accessed: 09.05.2024)

[20] Shmueli, G.; Bruce, P.C.; Gedeck, P. and Patel, N.R. (2020) *Data Mining for Business Analytics*. Hoboken: John Wiley & Sons, Inc.

[21] SSB (2024) *Eksport av laks*. Available at: https://www.ssb.no/statbank/table/03024/ (Accessed: 15.02.2024)

[22] Studenmund, A.H. (2017) *A Practical Guide to Using Econometrics*. 7th ed. Pearson Education Limited.

[23] yahoofinance (2024) *EUR/NOK (EURNOK=X)*. Available at: https://finance.yahoo.com/quote/EURNOK%3DX/history?period1=962409600&period2=1710201600&interval=1wk&filter=history&frequency=1wk&includeAdjustedClose=true&guccounter=1 (Accessed: 15.02.2024)

[24] Yoo, T.W. and Oh, L.S. (2020) Time Series Forecasting of Agricultural Products' Sales Volumes Based on Seasonal Long Short-Term Memory, *Applied Science*, 10(22), pp. 8169. Available at: https://doi.org/10.3390/app10228169 (Accessed: 09.05.2024)

[25] Yusop, Z.B.M. (2018) PESTEL analysis, *1st National Conference on Multidisciplinary Research and Practice 2018* Kuala Lumpur, 24.11.2018, pp. 34-39.

[26] Zitti, M. (2023) Forecasting salmon market volatility using long-short term memory (LSTM). *Aquaculture Economics & Management*, 28(1), 143–175. AVailable at: https://doi.org/10.1080/13657305.2023.2255346 (Accessed: 09.05.2024)