Retail demand forecasting using CNN- LSTM model

Nithin Soundar S J
Electronics and Communication Engineering,
Sri Ramakrishna Engineering College,
Coimbatore, India.
E-mail:
nithinsoundar.1902132@srec.ac.in

Rajasekar T
Electronics and Communication Engineering,
Sri Ramakrishna Engineering College,
Coimbatore, India.
E-mail: rajasekar.t@srec.ac.in

Jayanthy S
Electronics and Communication Engineering,
Sri Ramakrishna Engineering College,
Coimbatore, India.
E-mail: jayanthy.s@srec.ac.in

Karthik K
Electronics and Communication Engineering,
Sri Ramakrishna Engineering College,
Coimbatore, India.
E-mail: karthik.1902102@srec.ac.in

Rithick Roshan R
Electronics and Communication Engineering,
Sri Ramakrishna Engineering College,
Coimbatore, India.
E-mail:
rithickroshan.1902156@srec.ac.in

***Abstract*—This paper proposes a deep learning model to predict the stock that would be required by a store in a particular period with the help of historic information such as past sales. This task could help a business to run smoothly and make sound decisions but are very hard to predict accurately. A CNN- LSTM (Convolutional Neural Network- Long Short-Term Memory Network) model is proposed to forecast retail demand. This model equips the Swish Activation Function. This works better than the traditional and most successful activation function ReLU (Rectified Linear Unit). Data from 10 stores each consisting of 50 items are taken as input. This proposed work has implemented various other models such as Multilayer Perceptron, Long Short-Term Memory cells, Convolutional Neural Networks to predict sales. The experiment results suggest using CNN- LSTM Model as it has considerably lower RMSE (Root Mean-Squared Error).**

Keywords: Convolutional Neural Networks (CNN), Long Short-Term Memory Cells (LSTM), Multilayer Perceptron, Demand Forecasting, Neural Networks, Tensor Flow, Swish Activation Function, ReLU Activation Function.

I. INTRODUCTION

With COVID-19 on our doorsteps, the retailers are having a hard time meeting demand and supply. Retailers must maximize sales without overstocking and choose an optimal price for products. These problems can be overcome by forecasting. But the question one must ask is not 'How much do I have to sell?' but 'What is the customer demand?'. By focusing on demand, retailers can cut costs by keeping the right number of stocks on their shelves.

This model aims to predict the optimal stock of a particular product at a particular store on a particular day. It is a time series analysis of the past historic data. The data set consists of 50 items across 10 stores over a span of 4 years.

Section 2 explores the related works, section 3 analyses the methodology, section 4 overviews the experimental results and section 5 concludes the work.

II. RELATED WORKS

Fatima Zohra Benhamida et al [2020] proposed an online platform called Stock & Buy to incorporate the forecasting algorithm. They have proposed a tool named Comb-TSB which spontaneously chooses the most accurate model amidst a set of models. They have also proposed a clustering-based approach for products with less or no sales history.

Anish Palkar et al [2020] proposed an LSTM based retail demand forecasting model. In this paper, they have compared many models such as LSTM and Support Vector Machines. Among various methods, the one with high accuracy is to be selected.

Karan Wanchoo [2019] implemented a comparison between a pair of machine learning models- Deep Neural Network and (GBM) Gradient Boosting Method for demand forecasting. The paper discusses the feasibility of these models for a univariate time series. This is because not all retailers have access to supply chain metrics to make a multivariate model.

Yue-fang Gao et al [2010] have proposed a system for retail demand forecasting based on neural networks and implemented using Visual Basic for Applications. The parameters can be auto-adjusted with respect to errors and the model is independent of the accuracy of the mathematical expressions. The results show that accuracy is considerably improved by the usage of this model.

Yue-Fang Gao et al [2009] have proposed a demand forecasting model based on a Neural network. It builds a neural network on the Holt-Winter's model. The accuracy is greatly improved by minimizing a cost function

Yin Yafeng et al [2008] have proposed a Genetic algorithm and Fuzzy Neural Network (FNN) model based on a back propagation algorithm. They have developed algorithms that generate new fuzzy rules to handle the fuzzy neural network model. To optimize the model Back Propagation Algorithm and Genetic Algorithm are used.

Nimai Chand Das Adhikari et al [2017] have discussed about various models that can be employed to predict demand. They have set forth a new procedure to preprocess data for quality assurance. They have compared various statistical forecast models, time series models and regression models.

This paper proposes a CNN- LSTM model for demand forecasting. Traditional ReLU activation function is replaced with Swish activation function to improve the accuracy. The former shows signs of better accuracy and lower RMSE. This type of model will be beneficial for real-world applications.

III. METHODOLOGY

Fig.1. Describes the workflow to predict the sales. The first step is to preprocess the collected data. The NaN cells are dropped.

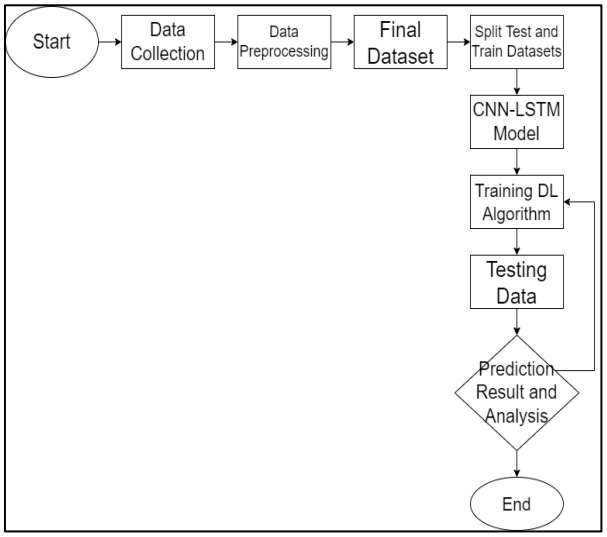

Fig. 1. Block Diagram

In the basic Exploratory data analysis, Figure. 2 shows the overall daily sales after aggregating the sales by day.

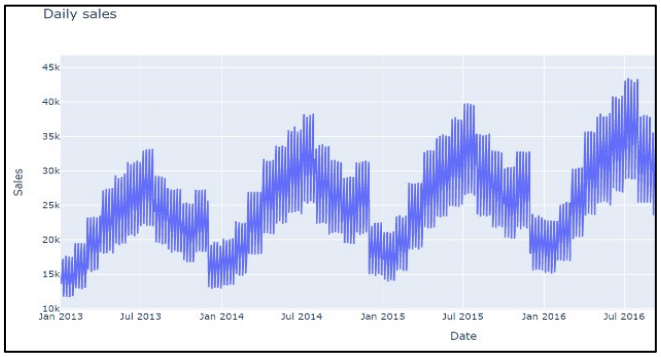

Fig. 2. Daily Sales

Prior to being modelled with the algorithm, the time-series algorithms ought to be transformed into supervised machine learning problems, i.e., a sequence of lists should be converted to combinations of input and output values. A time series can be assumed as columns of ordered values whereas a supervised machine learning problem consists of input and output values that a model can work on.

The data is then split into train and test data with train size as 60% and test size as 40% of original data.

A. *Multilayer Perceptron Model*

The input features of the Multilayer Perceptron (MLP) model is equal to the window size. MLP model doesn't take the input as sequence data, this may pose a problem since the model won't be able to analyse the data with the sequence pattern. The input is in a 2-D shape consisting of samples, timesteps

B. *Convolution Neural Network Model*

The CNN model will use two layers. The first is a 1-D convolutional layer and the second is a max-pooling layer. After passing through these layers, the output is then flattened to be interpreted by a hidden Dense layer and then forecasting the demand. The 1-D CNN layer identifies the patterns between the timesteps. The input is in 3-D shape consisting of samples, timesteps and features. The data is preprocessed to reshape and resample from 2-D data in the form of [samples, timesteps] into 3-D data in the form of [samples, timesteps, features]. The same 3-D data will be used for the LSTM model.

C. *Long Short Term-Memory Network Model*

The LSTM model perceives the input data as a sequence, unlike the MLP model. The model will try to learn the patterns, especially those patterns from sequences that are long.

D. *CNN-LSTM Model*

The LSTM model perceives the input data as a sequence, unlike the MLP model. The model will try to learn the patterns, especially those patterns from sequences that are long. The CNN- LSTM model fuses the merits of both CNN and LSTM models to predict sales. The CNN layer is used to extract time features and LSTMs predicts the sequence.

The CNN- LSTM algorithm was originally developed to predict spatial time-series data such as photos and videos. Traditionally called Long-term Recurrent Convolutional Network, the CNN-LSTM model can also be applied for problems with the 1-D structure of words.

Figure. 3 shows the working of the CNN- LSTM model.

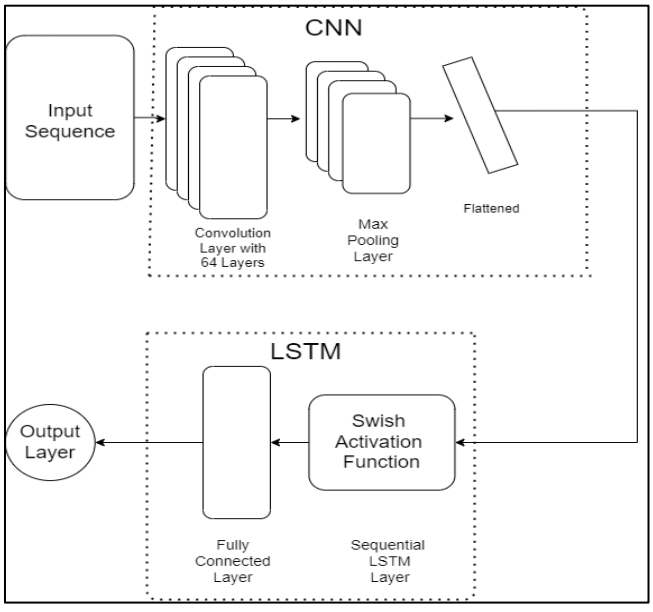

Fig. 3. CNN-LSTM Model

The general architecture of the CNN model always has multiple convolutional and pooling layers piled up after each other. The first layer of the algorithm is a 1-D convolutional network layer. This layer abridges the features in the data. Then input sequence is passed on to the second layer. This particular type of pooling layer, the max pooling layer, outputs a feature map with the most pre-eminent feature of the preceding feature maps. After the second layer, the feature maps are flattened. This is done in order to feed

into the LSTM layer. The model contains one LSTM layer which comes before a dense layer. The dense layer provides the output. The data is reshaped and rescaled to fit the 3-Dimensional input requirements. The input shape is 15-time steps with one feature.

Figure. 4. shows the detailed summary of the model.

```
_cnn_lstm = Sequential()
_cnn_lstm.add(TimeDistributed(Conv1D(filters=6
_cnn_lstm.add(TimeDistributed(MaxPooling1D(poo
_cnn_lstm.add(TimeDistributed(Flatten()))
_cnn_lstm.add(LSTM(50, Activation(custom_activ
_cnn_lstm.add(Dense(1))
_cnn_lstm.compile(loss='mse', optimizer=adam)
_cnn_lstm.summary()
```

```
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 time_distributed_3 (TimeDis  (None, None, 15, 64)     128
 tributed)

 time_distributed_4 (TimeDis  (None, None, 7, 64)      0
 tributed)

 time_distributed_5 (TimeDis  (None, None, 448)        0
 tributed)

 lstm_2 (LSTM)               (None, 50)                99800

 dense_6 (Dense)             (None, 1)                 51

=================================================================
Total params: 99,979
Trainable params: 99,979
Non-trainable params: 0
```

Fig. 4. Model Summary

The model is fit for 40 epochs and the loss was calculated using the mean squared error loss function.

E. Activation Function

Activation Function is one of the important mechanisms in neural networks that helps to decide whether the neuron should be activated or not. There are many popular activation functions. This paper has compared ReLU (Rectified Linear Unit) activation function and Swish Activation function.

1) ReLU Activation Function:.

ReLU is a non-linear activation function. It only deactivates neurons when the output is zero. This is where ReLU gives an edge over the other activation functions. Fig 5 represents the ReLU function plot.

f(y)=max (0, y)

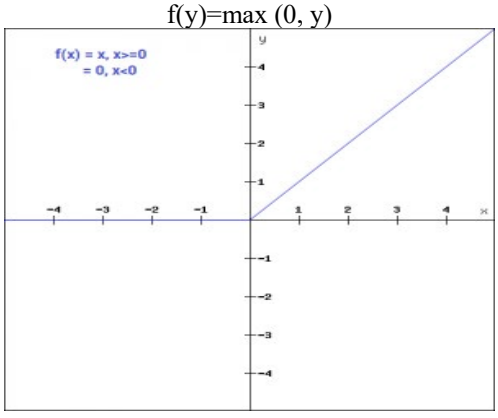

Fig. 5. ReLU Function

2) Swish Activation Function:

Swish Activation Function is a lesser popular function that was founded by Google. It is more efficient and shows better performance than ReLU. Fig 6 represents the swish function plot.

f(y) = y. sigmoid(y)

f(y) = y/(1-e^-y)

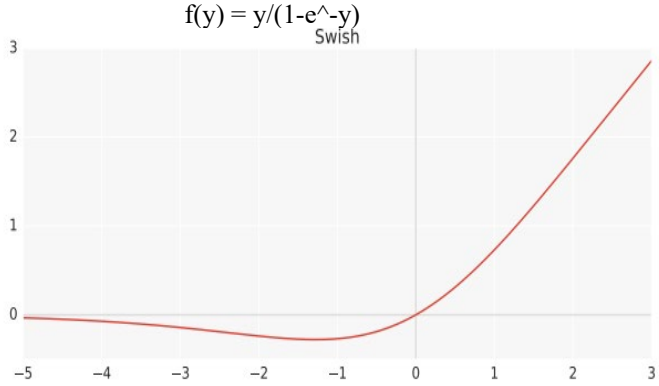

Fig. 6. Swish Function

IV. Experimental Results

The historic sales data set comprises more than 9,00,000 observations which are organized by date, store and item. The dataset was taken from the public data set available at Kaggle under a project competition to predict future sales. The columns of the dataset contain the date, store, item, sales. The test dataset consists of 40% of the datasets and the training dataset consists of 60% of the datasets. The proposed model was executed using an open-source platform called Google Colaboratory.

Table. 1 represents a few rows of the sales data. These data consist of 50 items across 10 stores from 2013 to 2017 i.e., 4 years.

TABLE I. Sample Train Dataset

	date	store	item	sales
48	27-05-2013	1	1	7
49	28-05-2013	1	1	12
50	29-05-2013	1	1	10
51	30-05-2013	1	1	19
52	31-05-2013	1	1	27
53	01-06-2013	1	1	26
54	02-06-2013	1	1	22
55	03-06-2013	1	1	12
56	04-06-2013	1	1	15
57	05-06-2013	1	1	24
58	06-06-2013	1	1	9
59	07-06-2013	1	1	21
60	08-06-2013	1	1	20
61	09-06-2013	1	1	38
62	10-06-2013	1	1	20
63	11-06-2013	1	1	18
64	12-06-2013	1	1	22
65	13-06-2013	1	1	21
66	14-06-2013	1	1	22
67	15-06-2013	1	1	26
68	16-06-2013	1	1	23

Figure 7, 8, 9 represents the loss plot for the CNN- LSTM model with ReLU Function, Swish function (40 Epochs), Swish Function (20 Epochs) respectively.

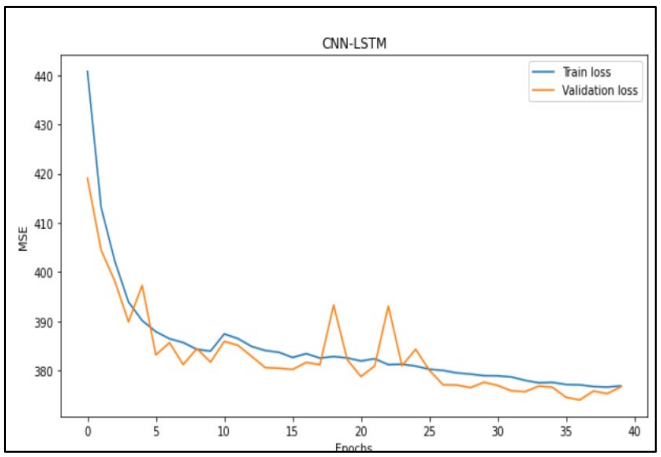

Fig. 7. Loss plot of CNN- LSTM with ReLU function

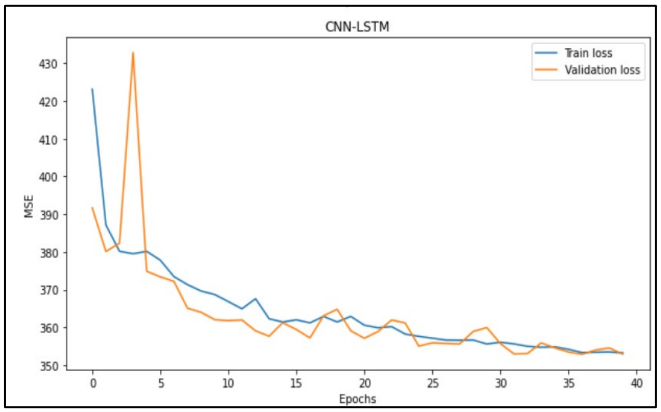

Fig. 8. Loss Plot of CNN- LSTM with Swish Function (40 Epochs).

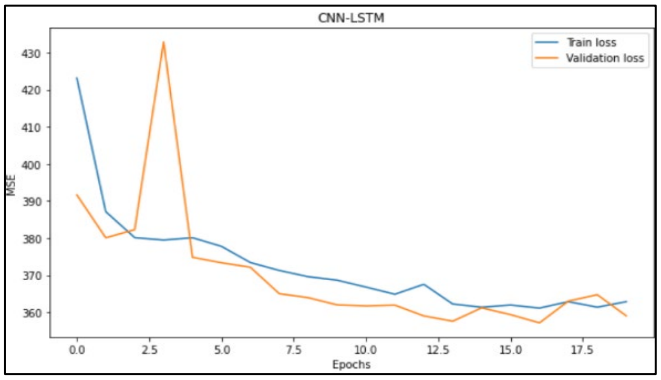

Fig. 9. Loss Plot of CNN- LSTM with Swish Function (20 Epochs).

Fig. 10 shows the losses using CNN- LSTM. Once the model is built, the model is trained using the fit () function and predicted using predict () function. The model gave out fewer losses with 40 Epochs and a batch size of 256

```
Epoch 1/40
3149/3149 - 19s - loss: 423.1304 - val_loss: 391.6559 - 19s/epoch - 6ms/step
Epoch 2/40
3149/3149 - 16s - loss: 387.1443 - val_loss: 380.1128 - 16s/epoch - 5ms/step
Epoch 3/40
3149/3149 - 19s - loss: 380.1570 - val_loss: 382.2972 - 19s/epoch - 6ms/step
Epoch 4/40
3149/3149 - 16s - loss: 379.5138 - val_loss: 432.8689 - 16s/epoch - 5ms/step
Epoch 5/40
3149/3149 - 17s - loss: 380.1377 - val_loss: 374.8536 - 17s/epoch - 5ms/step
Epoch 6/40
3149/3149 - 17s - loss: 377.8037 - val_loss: 373.3839 - 17s/epoch - 5ms/step
Epoch 7/40
3149/3149 - 18s - loss: 373.4488 - val_loss: 372.1870 - 18s/epoch - 6ms/step
Epoch 8/40
3149/3149 - 17s - loss: 371.3137 - val_loss: 365.0605 - 17s/epoch - 5ms/step
```

Fig. 10. Model fit

Table 2 illustrates the juxtaposition of RMSE between the Swish function and ReLU for different algorithms that are analyzed. Various metrics such as training RMSE, validation RMSE is inferred. By Comparing with the ReLU activation function, the CNN- LSTM model with the Swish function gives fewer losses.

Table 3 compares the loss for the Swish Activation function with 20 and 40 Epochs.

Table 4 compares the R squared value for the Swish function and ReLU. The swish function gives a higher R squared value than ReLU, thus the analysis infers that the CNN- LSTM model that equips the Swish function is considerably better as larger R squared values imply smaller differences between the observed data and fitted value.

S. No	Model	Swish		ReLU	
		Train RMSE	*Validation RMSE*	*Train RMSE*	*Validation RMSE*
1	**CNN-LSTM**	**18.39**	**18.39**	19.41	19.40
2	Multilayer Perceptron	18.31	18.40	18.31	18.44
3	CNN	18.45	18.58	18.56	18.66
4	LSTM	18.19	18.58	21.12	18.66

TABLE II. COMPARISON OF TRAINING RMSE AND VALIDATION RMSE WITH RELU ACTIVATION FUNCTION AND SWISH ACTIVATION FUNCTION

S. No	Model	Swish (40 Epochs)		Swish (20 Epochs)	
		Train RMSE	*Validation RMSE*	*Train RMSE*	*Validation RMSE*
1	**CNN-LSTM**	**18.39**	**18.39**	18.90	18.94
2	Multilayer Perceptron	18.31	18.40	18.41	18.53
3	CNN	18.45	18.58	18.58	18.68
4	LSTM	18.19	18.58	18.29	18.68

TABLE III. COMPARISON OF TRAINING RMSE AND VALIDATION RMSE WITH SWISH ACTIVATION FUNCTION OF 40 EPOCHS AND 20 EPOCHS

S. No	Model	Swish		ReLU	
		Train R^2	*Validation R^2*	*Train R^2*	*Validation R^2*
1	**CNN-LSTM**	**0.696**	**0.698**	0.623	0.620
2	Multilayer Perceptron	0.664	0.657	0.664	0.657
3	CNN	0.659	0.652	0.655	0.649
4	LSTM	0.668	0.652	0.553	0.649

TABLE IV. COMPARISON OF TRAINING R^2 AND VALIDATION R^2 WITH RELU ACTIVATION FUNCTION AND SWISH ACTIVATION FUNCTION

V. CONCLUSION

Predicting the demand for retail products is a challenging task as these depend on multiple parameters that form complex patterns. Nonetheless, this model has obtained results regardless of these complex patterns. The model was fed data consisting of historic sales data which was pre-processed. Various algorithms were compared and the best fit model was obtained. For future works, deep learning models which consider other parameters can be implemented.

REFERENCES

[1] Y. -F. Gao, Y. -S. Liang, Ying Liu, S. -B. Zhan and Z. -W. Ou, "A neural-network-based forecasting algorithm for retail industry," 2009 International Conference on Machine Learning and Cybernetics, 2009, pp. 919-924, doi: 10.1109/ICMLC.2009.5212392.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] Yuefang Gao, Yongsheng Liang, Fei Tang, Zhiwei Ou and Shaobin Zhan, "A demand forecasting system for retail industry based on neural network and VBA," 2010 Chinese Control and Decision Conference, 2010, pp. 3786-3789, doi: 10.1109/CCDC.2010.5498506.K. Elissa, “Title of paper if known,” unpublished.

[3] A. Palkar, M. Deshpande, S. Kalekar and S. Jaswal, "Demand Forecasting in Retail Industry for Liquor Consumption using LSTM," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 521-525, doi: 10.1109/ICESC48915.2020.9155712.

[4] Yin Yafeng, Liu Yue, Gao Junjun and Tan Chongli, "A new fuzzy neural networks model for demand forecasting," 2008 IEEE International Conference on Automation and Logistics, 2008, pp. 372-376, doi: 10.1109/ICAL.2008.4636178

[5] F. Z. Benhamida, O. Kaddouri, T. Ouhrouche, M. Benaichouche, D. Casado-Mansilla and D. López-de-Ipiña, "Stock&Buy: A New Demand Forecasting Tool for Inventory Control," 2020 5th International Conference on Smart and Sustainable Technologies

(SpliTech), 2020, pp. 1-6, doi: 10.23919/SpliTech49282.2020.9243824.

[6] J. S, C. R, Y. Y M and S. D, "Automatic Warning System for Drivers using Deep Learning Algorithm," 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), 2021, pp. 1730-1737,

[7] K. Wanchoo, "Retail Demand Forecasting: a Comparison between Deep Neural Network and Gradient Boosting Method for Univariate Time Series," 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), 2019, pp. 1-5, doi: 10.1109/I2CT45611.2019.9033651.

[8] Silaparasetti, Teja & Das Adhikari, Nimai & Domakonda, Nishanth & Garg, Rajat & Gupta, Gaurav. (2017). An Intelligent Approach to Demand Forecasting.

[9] https://www.kaggle.com/c/competitive-data-science-predict-future-sales/data

[10] https://machinelearningmastery.com/cnn-long-short-term-memory-networks/

[11] https://www.kaggle.com/dimitreoliveira/deep-learning-for-time-series-forecasting

[12] https://www.analyticsvidhya.com/blog/2021/05/know-the-best-evaluation-metrics-for-your-regression-model/

[13] https://medium.com/@danqing/a-practical-guide-to-relu-b83ca804f1f7#:~:text=ReLU%20stands%20for%20rectified%20linear,max(0%2C%20x).&text=ReLU%20is%20the%20most%20common,usually%20a%20good%20first%20choice.

[14] https://medium.com/@neuralnets/swish-activation-function-by-google-53e1ea86f820

[15] https://towardsdatascience.com/swish-booting-relu-from-the-activation-function-throne-78f87e5ab6eb

[16] https://towardsdatascience.com/get-started-with-using-CNN- LSTM-for-forecasting-6f0f4dde5826