**Holiday Forecasting: A Machine Learning Approach to Forecast Shipment Volume for a Leading Fashion Retailer**

Arjun Chakraborty, Hemanth Devavarapu, Vijaya Rani, Almas Syed Rizvi, Matthew A. Lanham

Purdue University, Department of Management 403 W. State Street, West Lafayette, IN 47907 chakra38@purdue.edu, hdevavar@purdue.edu, vrani@purdue.edu, rizvi7@purdue.edu, lanhamm@purdue.edu

**ABSTRACT**

This study provides an approach to predict the estimated number of packages for various origin destination combinations during the holiday season. The motivation for this study is that retailers need accurate demand forecasts for tactical and operational planning within the business. Generating accurate forecasts can be a challenging endeavor, but even more so during holidays, where chaotic demand spikes occur. Having the ability to plan reliably for these special days during a holiday season can give visibility to third party logistics partners about the right amount of shipments being shipped to customers. In collaboration with a national retailer we researched and developed a methodology that can capture these unusual holiday shipment volume peaks using machine learning algorithms, then estimate the number of packages for various origin destination combinations. We found that SARIMA and Long Short-Term Memory (LSTM) neural network models can be used to model the time series to estimate shipment volume based on the previous two years holiday season shipment patterns.

Keywords: Time Series Forecasting, Holiday Season, Transportation Planning, Neural Networks, Machine Learning Algorithms

**INTRODUCTION**

Generating accurate forecasts can be a challenging endeavor, but even more so during holidays, where chaotic demand spikes occur. Forecasting the volume of holiday shipments would help the retailers give more visibility to their logistic partners to plan last mile operations efficiently. Efficient capacity planning has been the core of last mile supply chain which leads to better negotiation for rates with partners and help in tactical and operational planning within a Fulfilment Centre. By utilizing these forecasting models in the last mile planning, companies are seeing more efficient workflows in every step of logistical transition: from analysis, planning, dispatching, tracking and reporting. With the variety of routing needs, having specialized workflows or process geared to adapt to fixed, dynamic or hybrid routing is imperative. By having the most advanced machine learning algorithms, planning can be more accurate with less manual routing: reducing empty miles, improving capacity usage, optimizing manager and driver time and generating cost savings. The objective is to assign shipping load to different modes of transport and carriers such that the utilization of the carrier is maximized and transportation costs are minimized.

The 6-week period ranging from roughly Thanksgiving to New Year is by far the most important for retailers and often accounts for half of the whole-year sales. It is a massive peak period for retailers and for this reason the company needs to start planning paid media, IT and logistics strategies as early as possible.

Indeed, from a retailer’s point-of-view, Christmas is not so much an event but rather a process: as soon as New Year celebrations are over, it is time to start planning for the next Christmas: What to sell, where to source it, and how much and when to buy. These questions need to be answered long before Christmas; how long depends on the lead times of the products in question. Just before the season supply chain managers must ensure that products end up in shelves in time and in correct quantities and they must ramp up their last mile accordingly for on time delivery.

Some of the research question which stems to the retail industry is that when does the season start and when does it finish exactly? Is it all about Black Friday and Cyber Weekend, or rather, it is about December Christmas gifts? Does it start in early November or late November just before Thanksgiving? Does it last even after Christmas Day and continue into New Year? Maybe the online traffic ramp-up starts earlier in the season than the revenue one, as people begin to surf around and decide to buy just at a second stage? Are people concerned about faster service? Are people ready to pay premium for faster delivery before Christmas?

Building an accurate forecasting model by the industry could enable them to solve some of the problems. In this paper, we select and review a set of papers from the literature. In order to have a comprehensive collection of papers, we employed the popular and powerful research portals of  <http://scholar.google.com/> and  <http://www.sciencedirect.com/> and search objectively by keywords of “shipment forecasting” “neural networks for forecasting,” “machine learning approach to predict shipment volume”.

We organized the remainder of this paper as follows: A review of the literature on various criteria, different machine learning algorithms and methods used for forecasting is presented in the next section. Next, we examine the data, outline our methodology, and discuss the models we developed. Different models were evaluated on the basis of Mean Absolute Percentage Error as was required by the client. Lastly, we provide discuss our results and conclusions.

**LITERATURE REVIEW**

Forecasting holiday season demand such as those for fast fashion goods is a typical problem faced across industries, especially when demand is discount driven. Whenever there is a certain degree of uncertainty involving future outcomes, using a time series approach can yield useful insights that support decision making. With organizations understanding the need to anticipate future outcomes, developing accurate forecasts have been one of the top priorities over the last few years. Finally, chaotic demand with sporadic spikes poses challenges industry wide, and techniques needs to be improved to help companies efficiently address these concerns.

To comprehend how deep learning is especially beneficial for time series modeling, we found Längkvist, Karlsson et al. (2014) paper on unsupervised feature learning and deep learning for time-series modeling very useful. The paper offers insights into recent developments in deep learning and unsupervised feature learning for time-series problems. It also addresses the challenges present in the time series data and provides reviews of previous works which have applied this approach across a variety of forecasting problems and suggests certain modifications to these algorithms. Similarly, to understand how Long Short-Term Memory (LSTM) recurrent neural network (RNN) models can be used to make predictions Fischer and Krauss (2018) study on demonstrate this model for financial market predictions. Although the paper is based on the application of LSTM for financial time-series predictions, it also provides insights on time series predictions in general. It offers a comparative study of LSTM and methods and shows why LSTM is a superior technique for sequence learning.

To understand the working principles of time series decomposition and sequence modeling we found (Hyndman and Athanasopoulos 2018) a useful resource, and the extension of modeling time series using machine learning and deep learning (Goodfellow, Bengio et al. 2016) quite useful.

To understand how LSTM performed in comparison to other neural networks (Azzouni and Pujolle 2017) was particularly useful. The paper aims to develop a real-time time-series model that provides the flexibility of real-time monitoring. The authors have used several methods of time-series prediction such as linear prediction, Holt-Winters algorithm, and neural networks. Finally, LSTM RNN architecture was developed using the python Keras library and used for prediction. It was found that the LSTM RNN was the best prediction model for time-series.

For many different time-series datasets, having an idea and working knowledge of clustering in time-series forecasting would help us combine datasets based on factors such as industry type, kind of market etc. Toshniwal and Joshi (2005) examined clustering time-series data to obtain cumulative weighted slopes that could be used for feature extraction. Slopes were calculated at corresponding points of each of the time series. The slopes computed at corresponding points of the sequences were then assigned weights depending on the location of the slope along the time axis. Weighted slopes were obtained for each of the time sequences which were then summed to obtain the cumulative weighted slope for the respective time sequence. The cumulative weighted slopes were then grouped into clusters using k-means clustering method to identify similar patterns.

Finally, running an LSTM model, without having other models to compare the results with, one would not be able to conclude whether the model performed well or not. For this reason, Zaiyong Tang, Chrys de Almeida, Paul A. Fishwick’s paper on Time series forecasting using neural networks vs. Box- Jenkins methodology as well as JW Taylor’s paper on Short-term electricity demand forecasting using double seasonal exponential smoothing gave us an idea on other approaches such as ARIMA modeling and exponential smoothing.

|  |  |  |
| --- | --- | --- |
| **Author, Year** | **Study** | **Motivation for research** |
| (Hyndman and Athanasopoulos 2018) | (Book) Forecasting Principles and Practice | To comprehensively understand time series decomposition and various advanced forecasting methods |
| (Brown 2018) | Forecasting Time-Series data with Prophet | To gain an understanding of applications of Facebook’s Prophet in time series forecasting and challenges faced while using it. |
| (Fischer and Krauss 2018) | Deep learning with long short-term memory networks for financial market predictions | To gain insights into the prediction capabilities of LSTM. |
| (Azzouni and Pujolle 2017) | A Long Short-Term Memory Recurrent Neural Network Framework for Network Traffic Matrix Prediction | To study LSTM RNN in comparison with other time-forecasting methods |
| (Goodfellow, Bengio et al. 2016) | (Book) Deep Learning | To help build a mathematical background of relevant topics like machine learning, deep neural |
| (Karlsson 2013) | A review of unsupervised feature learning and deep learning for time-series modeling | To gain an understanding of applications of deep learning in time series forecasting and challenges faced while using it |
| (Toshniwal and Joshi 2005) | Using Cumulative Weighted Slopes for Clustering Time Series Data | To study a new approach for clustering time series data |
| ( Zaiyong Tang, Chrys de Almeida, Paul A. Fishwick 1991) | Time series forecasting using neural networks vs. Box- Jenkins methodology | To study the results of a comparative study of the performance of neural networks and conventional methods in forecasting time series. |

Table 1: Literature Review

**DATA**

The data used in our study comes from a leading American fashion retailer. Three years shipment level data at Fulfilment Center (FC) and zip-code level is provided, covering the holiday season of approximately six weeks. Holiday’s covered include Thanksgiving (17th/18th), black Friday/cyber Monday through Christmas and New Year. Additional factors like weight were provided along with the shipping zone of the customer and service mode of the delivery. A detailed data description is shown in Table 2.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field** | **Core Data** | **Data Type** | **Length** | **Comments** | **Field / Value** |
| Client Code (2) | Y | varchar2 | 4 |  | "NORD", "NIDR", "HAUT" |
| Manifest Date | Y | date |  | Shipment start date | Shipment Pick-up Date |
| Normalized Carrier (2) | Y | varchar2 | 20 | Delivery Carrier ID | "FedEx", "USPS", "UPS", "OnTrac", "UDS" |
| Normalized Location | Y | varchar2 | 3 | Partner location where shipping originates | "ECFC", "MWFC", "Store/RDR" |
| Shipper City | Y | varchar2 | 30 | City of source |  |
| Shipper State | Y | varchar2 | 2 | State of source |  |
| Shipper Zip Code | Y | varchar2 | 15 | Zip code of source |  |
| Recipient City | Y | varchar2 | 30 | City of destination | ShipToCity |
| Recipient State | Y | varchar2 | 2 | State of destination | ShipToStateCode |
| Recipient Zip | Y | varchar2 | 15 | Zip code of destination | ShipToZipCode |
| Normalized Service (2) | Y | varchar2 | 4 | 1Day, 2Day, 3Day and ground delivery | "1D", "2D", "3D", "GRND" |
| Ship Weight (2) | Y | number | 10,2 | weight of package |  |
| Zone |  | number | 10,2 | delivery area zone |  |
| Pieces | Y | number | 10,2 | number of packages |  |

Table 2: Data description

**EXPLORATORY DATA ANALYSIS**

The holiday season each year is selected as the last two weeks of November and the month of December. The holiday season for three years are concatenated to resemble a time series as follows. As shown in Figure 1, there are seasonal patterns in the sale during the holiday season. There is a severe dip in shipments during 2017 after 15th December which were not observed during 2016 and 2018.

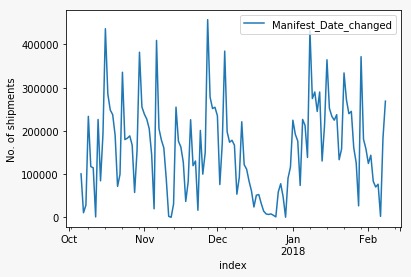


Figure 1: Daily level sales for 2016-2018

Figure 2 shows the demand by customers during holiday season across United States. The color gradient depicts the difference in demand across states. California, Texas, Illinois and New York constitute about 40% of the demand.

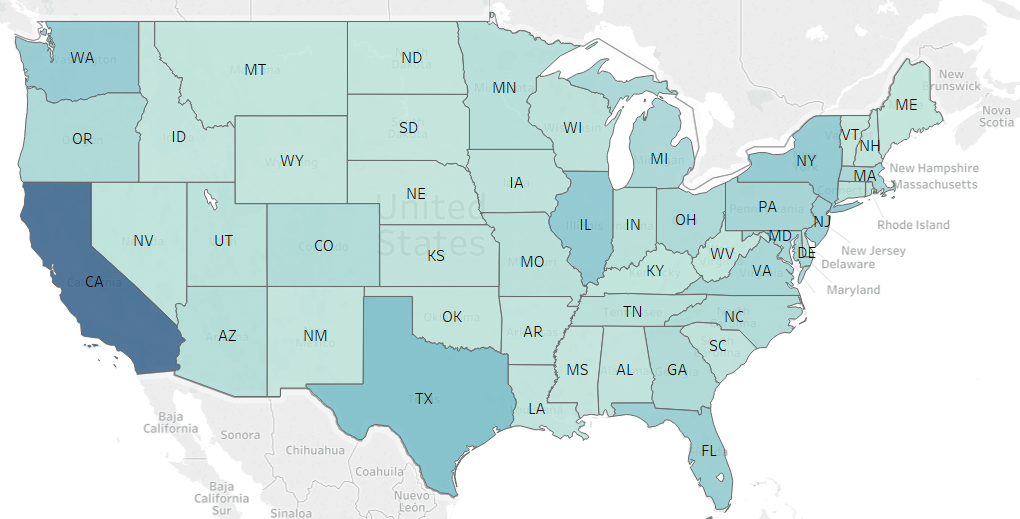
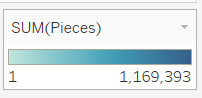
 

Figure 2: Demand distribution across United States

Figure 3, shows shipment weight across locations. It is evident that shipment weight remained constant (except a few peaks/outliers) for a given origin location across the holiday period. We used this insight to predict weight on the basis of its distribution at origin level.

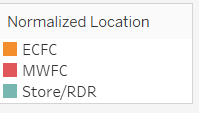
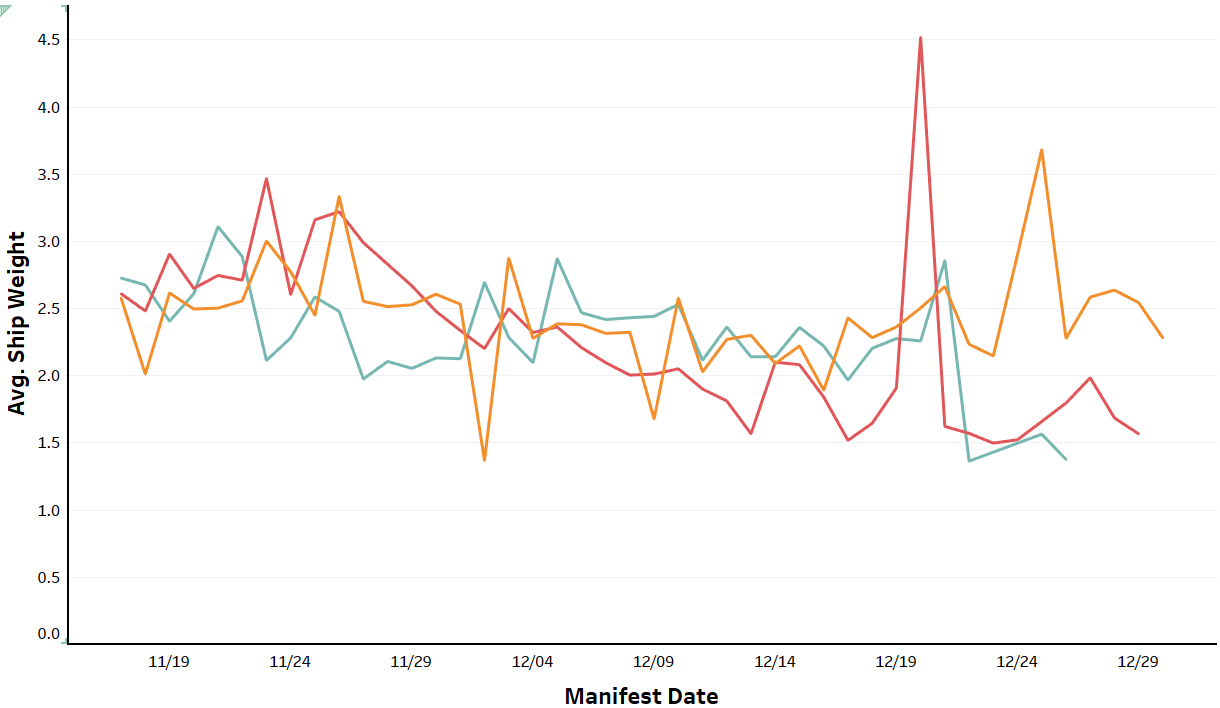


Figure 3: Average shipment weight across the normalized location

**METHODOLOGY**

Our study is to forecast the shipment volume for each destination at daily level. Source is the starting point of the shipment and in the case there are two Fulfilment Centers and number of Stores classified in one category. There are 68 zones and these zones are determined by the distance of destination from source. We approached it as a forecasting problem using the trends and seasonality of 204 source-zone combinations by building separate time-series. Figure 4 below outlines our methodological design.

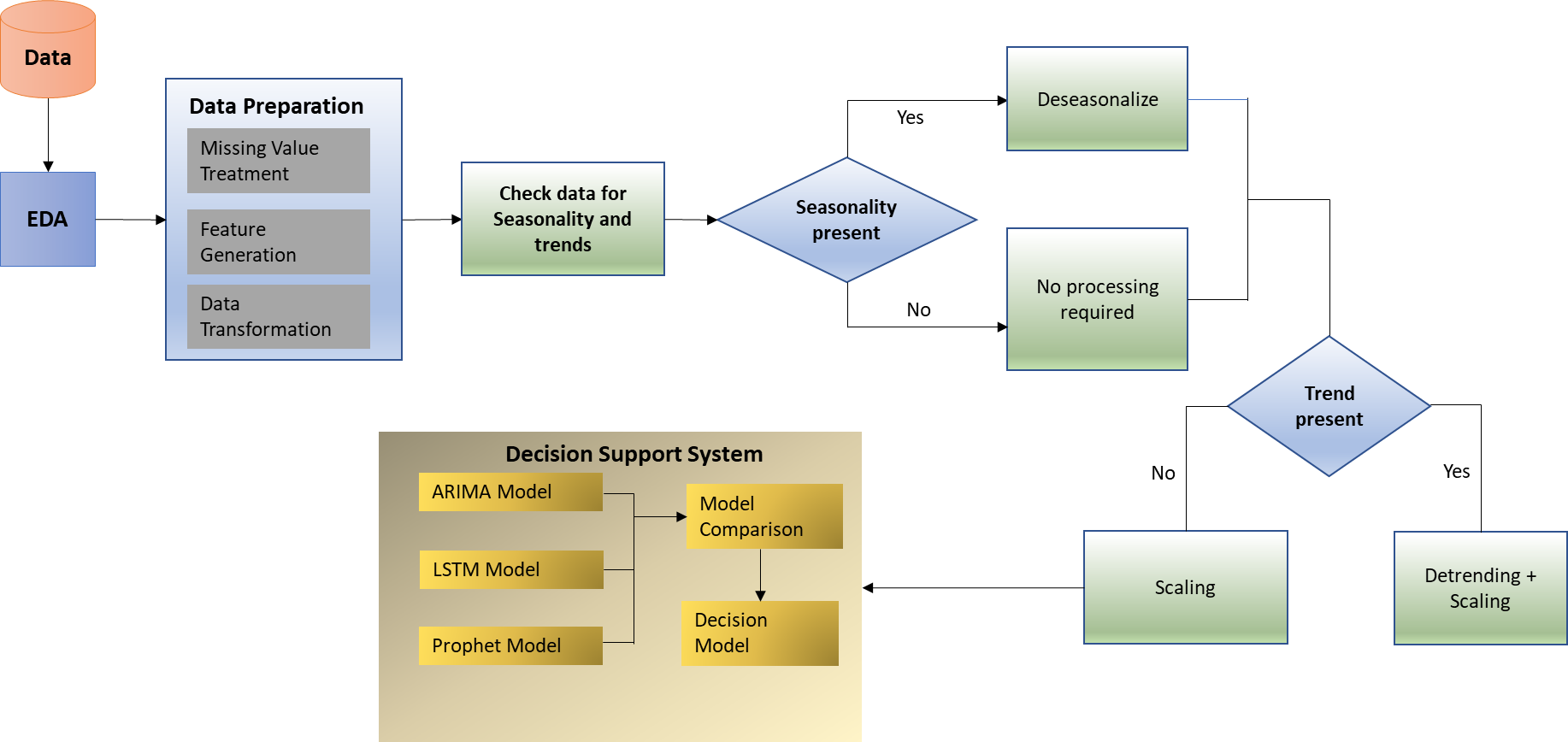
****

Figure 4: Forecasting methodology

**Data-cleaning**

* Removed observations with missing values in zone & zip-code
* Shipments with ‘0’ weight are considered in < 1 pound weight bucket
* Make 204 distinct combinations of fulfilment centre and zone
* Three years of data were brought in consecutive intervals where each year was having six weeks of shipment information

**ARIMA Model**

One of the most common methods used in time-series forecasting is known as the Autoregressive Integrated Moving Average (ARIMA) model. ARIMA is a model that can be fit to time-series data in order to better understand or predict future points in the series. There are three distinct parameters (p, d, q) that are used to parametrize ARIMA models. Because of that, ARIMA models are denoted with the notation ARIMA(p, d, q). Together these three parameters account for seasonality, trend, and noise in datasets:

* p is the auto-regressive part of the model. It allows us to incorporate the effect of past values into our model. Intuitively, this would be similar to stating that it is likely to be warm tomorrow if it has been warm the past three days.
* d is the integrated part of the model. This includes terms in the model that incorporate the amount of differencing (i.e. the number of past time points to subtract from the current value) to apply to the time-series. Intuitively, this would be similar to stating that it is likely to be same temperature tomorrow if the difference in temperature in the last three days has been very small.
* q is the moving average part of the model. This allows us to set the error of our model as a linear combination of the error values observed at previous time points in the past.

Seasonal ARIMA (SARIMA) was also fit to the time-series data for volume of shipments for various source-destination combinations and was used to forecast shipments for 2019. Seasonality factor was taken as 42 days, since this was observed from the exploratory data analysis. Exogenous factors like service delivery mode were taken into account while building the time-series model. 2018 holiday season data was shifted to the period after 2017 data to take into account continuity in data. Additive model was used as there was no correlation between trends and seasonality.

**LSTM Model**

The long short-term memory recurrent neural network (LSTM RNN) model is specifically designed to learn long term dependencies, overcoming the problems of vanishing and exploding gradient. The current model works on a many-in-many-out mechanism, meaning it predicts multiple forecast outputs using multiple inputs (lag variables).

LSTM tends to perform better than traditional and other advanced machine learning forecasting methods like ARIMA, Random Forest (bagged tree model), etc., because of its abilities to learn long-term dependencies, which is crucial in time-series modeling. One disadvantage of using neural networks is that it can be very hard to train the model, especially on smaller sets of data aggregated over years, quarters or months. With smaller datasets, you are more prone to overfit to the training data and hence no generalize to hold out data.

LSTM was first introduced by Sepp Hochreiter and Jürgen Schmidhuber and improved in 2000 by Felix Gers' team. LSTM networks are popularly used in speech recognition, handwriting recognition etc. Data handling and preparation is conducted in Python 3.6. Our deep learning LSTM networks were developed with Keras on a TensorFlow backend. The LSTM network is trained on a CPU cluster. LSTM networks have connections between units which form a directed cycle. This allows them to retain memory (i.e. exhibit temporary dynamic behavior). LSTM networks are capable of learning long-term dependencies and can overcome the previously inherent problems of RNNs, (i.e., vanishing and exploding gradients). LSTM networks, like dense layers, have an input layer, one or more hidden layers, and an output layer. The number of neurons in the input layer is equal to the number of explanatory variables (feature space).

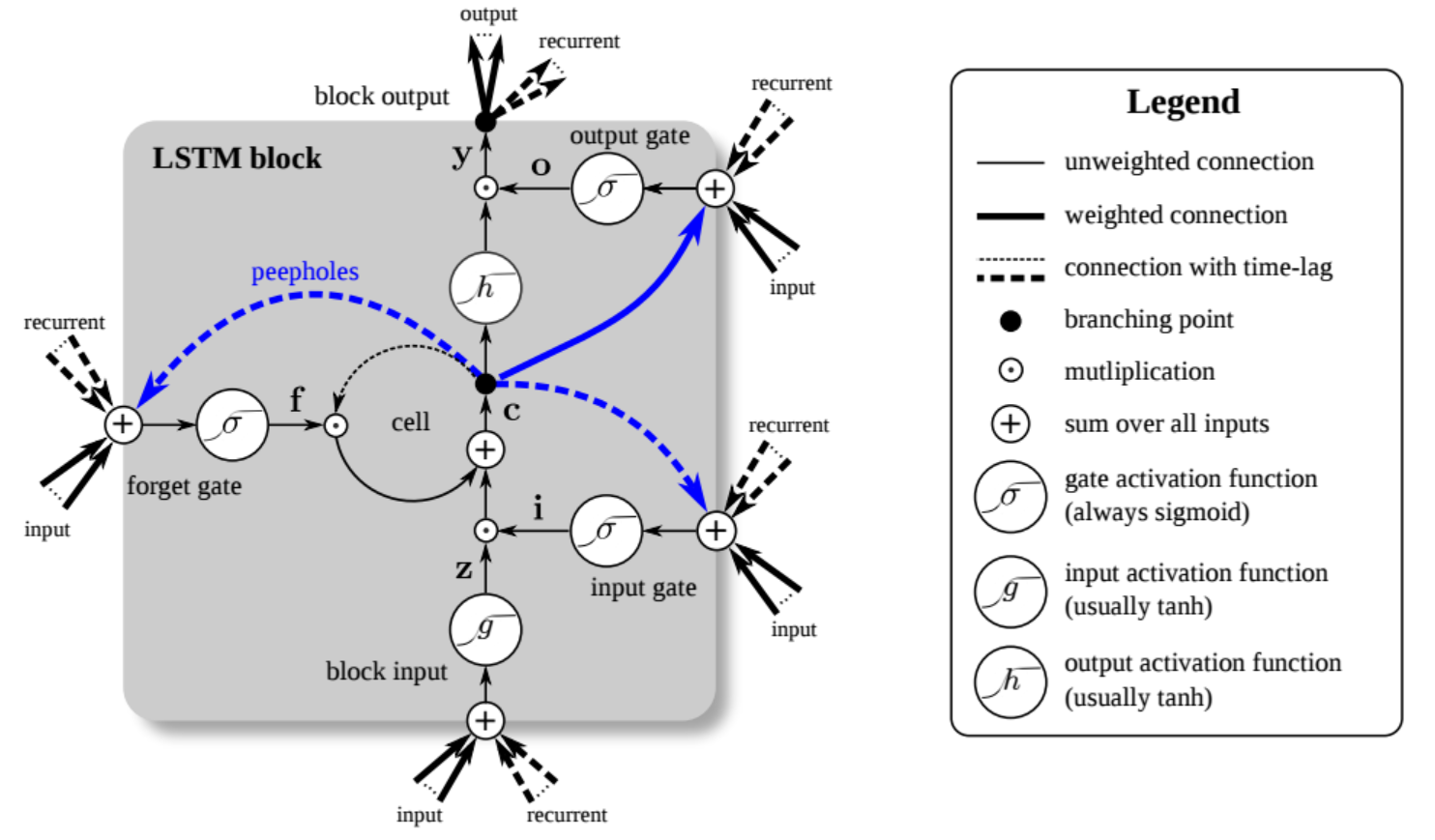


Figure 5: An LSTM Unit (Image by Klaus Greff2)

The output of the LSTM network is later inverse transformed to obtain the original range of values. Also, the seasonal and trend components are added back to the forecast output from the model. The performance of the LSTM model is judged over MAPE (Mean Absolute Percentage Error) across all separate time series.

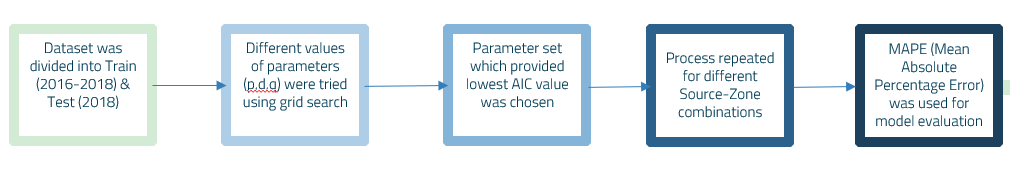


Figure 6: SARIMA approach

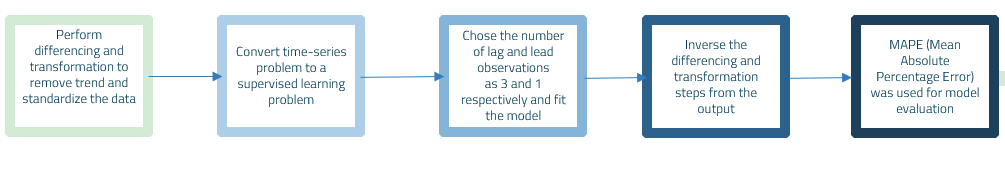


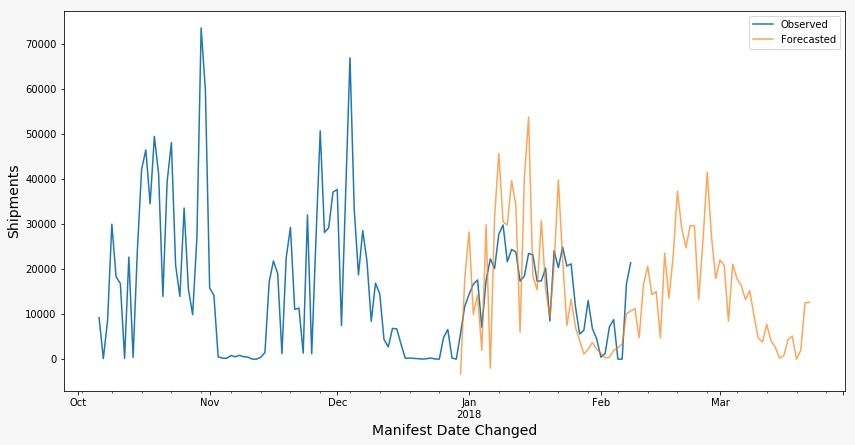
Figure 7: LSTM approach

The Mean Absolute Percent Error (MAPE) measures the size of the error in percentage terms. Many organizations focus primarily on the MAPE when assessing forecast accuracy. Most people are comfortable thinking in percentage terms, making the MAPE easy to interpret. It can also convey information when you don’t know the item’s demand volume. For example, telling your manager, "we were off by less than 4%" is more meaningful than saying "we were off by 3,000 cases," if your manager doesn’t know an item’s typical demand volume. Our clients also preferred MAPE as effective model evaluation technique to give preference to the model.

**RESULTS**

**SARIMA:**

Three years of data was used to train our model and the third year was used to validate/test it. This is because training the model on two years of data could not produce accurate forecasts. For validation, we are considering MAPE. Out of the 204 source-zone destinations, the top 25 contribute to 96% of the shipment volume. We predicted for the top 25 using separate time-series. Figure 7 is the prediction result for the top source-zone (MWFC-7)



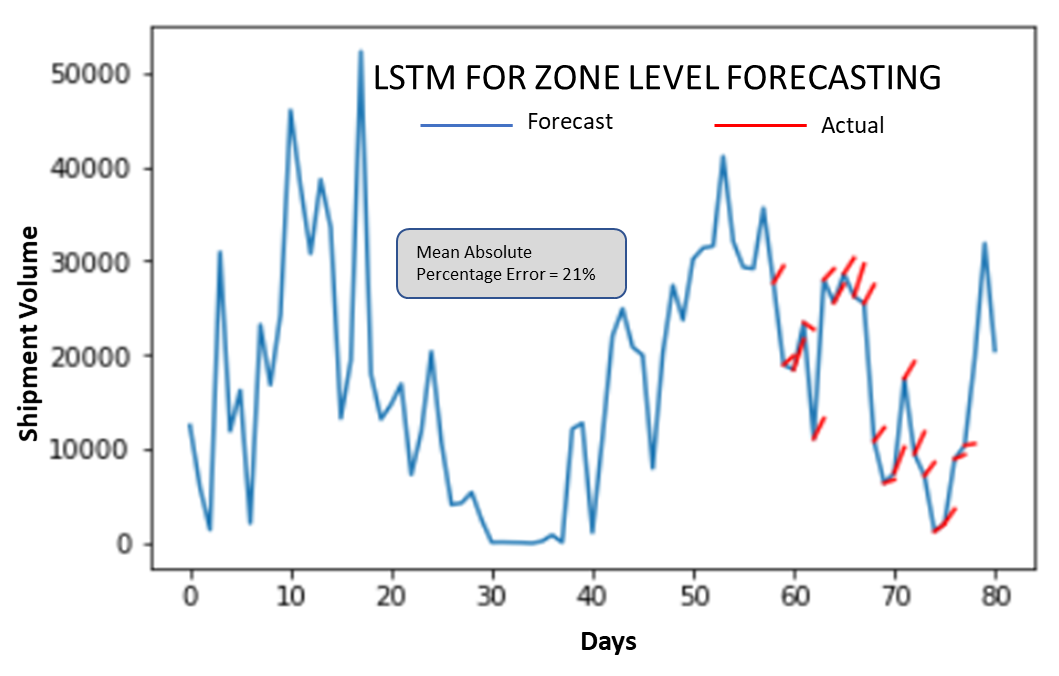
Mean Absolute Percentage Error = 10%

Figure 8: SARIMA forecast

In top 10 FC-Zone combinations, the best MAPE observed was 10% and the least was 46%.

**LSTM:**

LSTM was similarly modelled for the above time-series corresponding to the source–zone combinations and the average MAPE was found to be 70%. Figure 8 below shows the forecast on the validation window.



Mean Absolute Percentage Error = 70%

Figure 9: LSTM forecast

Table 3 shows the Mean Absolute percentage error for different zones. It is mainly varying between 10 and 46% barring one. These top zones account for 60% of the shipment volume.

|  |  |
| --- | --- |
| **Source-Zone Combination** | **MAPE %** |
| MWFC-7 | 29.89 |
| MWFC-5 | 11.59 |
| Store/RDR-1 | 134.97 |
| ECFC-2 | 10.26 |
| Store/RDR-2 | 26.02 |
| MWFC-3 | 46.64 |
| ECFC-8 | 11.21 |
| ECFC-4 | 11.72 |
| Store/RDR-4 | 19.16 |
| Store/RDR-5 | 20.04 |

Table 3: MAPE for Top 10 Zones

**CONCLUSION**

Holiday Planning is the core of operational performance in most industries and thus an efficient forecast can help organizations. It was observed that SARIMA is one such model that performed better (lower MAPE) than LSTM and aggregating shipments at weekly level provided better accuracy than that at daily level. However, with more data points in each time series the LSTM and SARIMA will train better and the accuracy of forecasts will be enhanced. Also incorporating more years of data would have enabled these models to give a much better accuracy.

**REFERENCES**

Azzouni, A. and G. Pujolle (2017). "A long short-term memory recurrent neural network framework for network traffic matrix prediction." arXiv preprint arXiv:1705.05690.

Brown, E. (2018). "Forecasting Time Series data with Prophet." Retrieved 3/1/2019, 2019, from <https://pythondata.com/forecasting-time-series-data-prophet-part-4/>.

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

Goodfellow, I., et al. (2016). Deep learning, MIT press.

Hyndman, R. J. and G. Athanasopoulos (2018). Forecasting: principles and practice, OTexts.

Karlsson, R. (2013). "Data as Intelligence–A Study of Business Intelligence as Decision."

Längkvist, M., et al. (2014). "A review of unsupervised feature learning and deep learning for time-series modeling." Pattern recognition letters **42**: 11-24.

Tang, Z., et al. (1991). "Time series forecasting using neural networks vs. Box-Jenkins methodology." Simulation **57**(5): 303-310.

Toshniwal, D. and R. Joshi (2005). "Using Cumulative Weighted Slopes for Clustering Time Series Data." International Transactions on Computer Science and Engineering **20**: 29-40.

Zaiyong Tang, Chrys de Almeida, Paul A. Fishwick (1991). “Time series forecasting using neural networks vs. Box- Jenkins methodology”