

Short-Term Passenger Count Prediction for Metro Stations using LSTM Network

Sminu Izudheen^a, Jaison Paul Mulerikkal^b, Mahima Jojee John^c, Malavika K^d, Jesmin Joshy^e and Gabriel M Beveira^f

^aAssociate Professor, Department of Computer Science and Engineering, Rajagiri School of Engineering and Technology, Kochi 682039, India.

^bProfessor, Department of Information Technology, Rajagiri School of Engineering and Technology, Kochi 682039, India.

^{c,d,e,f}Research Scholar, Department of Computer Science and Engineering, Rajagiri School of Engineering and Technology, Kochi 682039, India.

Article History: Received: 10 November 2020; Revised 12 January 2021 Accepted: 27 January 2021; Published online: 5 April 2021

Abstract: Predicting passenger flow is vital for the management, safety and smooth operation of any metro station. Such predictions are highly challenging as it depends on many parameters including travel pattern of the passengers. In this paper, we propose a highly efficient Long Short Term Memory Network [LSTM] which is a specialization of RNN to achieve this task. To do this prediction we employ the historical dataset from the metro containing the count, age and gender category of the passengers. Unlike earlier works, we also take into account the meteorological data of that time period and also the holiday information which includes the local events and public holidays. This accounts for the occasional spikes or fluctuations in the crowd patterns. Also the information about gender and age category of passengers is given emphasis and considered as an important parameter that affects the overall passenger count. Various configurations of the LSTM model are experimented by training the model repeatedly and the ones that yield the best result for this problem are evaluated and analyzed. The results obtained can be used to build an accurate and reliable predictive model to understand beforehand the amount of passenger crowd to expect

Keywords: Long short-term memory, Passenger Volume Prediction, Recurrent Neural Network

1. Introduction

A metro rail is a thread that seamlessly connects the city together and provides a hassle-free travel experience to every commuter. Being able to know beforehand the crowd of passengers that can be expected on a certain day is vital to the operation of the metro. Knowing this information can help the metro authorities to stay prepared beforehand to efficiently handle the rush. On a day when more crowds are expected, more staff can be assigned at that particular station and more ticketing counters and security check personnel can be employed. This knowledge helps to effectively deploy the existing staff and resources between the various metro stations of the city without having to invest more capital to manage the occasional peak hours. This will help to prevent crowd congestion, enhance safety of the passengers and help in the intelligent utilization of the existing metro capacity. Knowing the passenger flow patterns can be used to assign more autos and cabs at the busier stations and during the peak hours for further commutation of the passengers. This eases the travel experience of the common man through the city.

A public transport system like the metro is also a commercial hub. It provides a plethora of business opportunities if used rightly. This is where the need to know the gender and age category of passengers and their distribution throughout the week or functional hours of the metro throughout the day comes in. This can be used to display advertisements that target the appropriate crowd. It can also be used in the planning of various shops and complexes near the metro stations that are relevant to the various age and gender category of the crowd coming there. In this paper the Long Short term Memory Model (LSTM) is used, which is a modified version of the Recurrent Neural Network (RNN) with an emphasis on the past data in the memory. Here, future trend is predicted using historical data that has the gender and age details of the metro passengers on an hourly basis. Though there have been previous works in this area, the dependence on meteorological data over that time period and the holiday information of that day were not given sufficient emphasis. Also, in this paper we consider the gender and age category to which the passengers belong as a parameter affecting the prediction. However, this prediction poses a challenge because the data employed here, has no trends or linearity in its basic underlying structure and hence the traditional methods are inefficient here. Different aspects and configurations of the LSTM model were analysed and the one that gives the best result for the problem is selected.

2. Related Works

There have been considerable efforts in the domain of prediction in diverse fields and for various purposes using the efficient LSTM network. An attention-based LSTM model was used for forecasting the stock prices [1]. It denoised historical stock data, extracted and trained its features, and established a prediction model. Prediction of rainfall was done using Intensified Long Short-Term Memory (Intensified LSTM) network [2]. But it was found that the proposed intensified LSTM, when compared with the existing LSTM model did not show a significant improvement in accuracy but preserved the accuracy for further number of epochs. Tourism flow prediction was done using LSTM network and led to better performance as compared to Auto Regressive Integrated Moving Average (ARIMA) and Back Propagation Neural Network (BPNN) [3]. Reservoir inflow forecasting was done using ANN and various architectures of the RNN model [4]. It was found that RNN, particularly the LSTM network gave best prediction results and higher accuracy. Also there have been various initiatives and study done in the domain of passenger flow and crowd flow prediction. These used a variety of methods for crowd prediction. A model based on pyramidal Convolutional Recurrent Network was used for crowd density prediction [5]. Recurring periodic patterns were leveraged and a weighting based fusion mechanism was used to take into account the relevance of periodic representation. A spatio-temporal U-shape network model (ST-Unet) was proposed for station level crowd flow forecasts [6]. It focussed on spatio-temporal dependence. However, its prediction accuracy was low especially on rainy/foggy days and holidays. The passenger inflow and outflow volumes were predicted using a spatiotemporal graph convolutional neural network [7]. But these predictions led to abnormally high errors for certain stations.

A comparative analysis was done between the classical predictive models for predicting subway passenger flow where the support vector regression model and multilayer perceptron neural network model yielded best results when examined under normal and anomalous traffic conditions [8]. A probabilistic model selection method for metro flow prediction was done [9]. Here, passenger flow patterns were extracted using origin-destination information for the smart-card data.

A 'n-day' moving average passenger flow volume based on ARIMA (Auto-Regressive Integrated Moving Average) model for forecasting daily passenger flow was done [10]. The model emphasized the representative and stability of data sequences. Subway passenger flow prediction using the ARIMA model was developed, focusing on selecting the most appropriate parameters - p and q through Stationarity Test such as Dickey-Fuller test [11]. But ARIMA and its variants requires statistical properties of data to remain constant, making it applicable only to stationary time series data.

Various research and studies in the prediction domain have proved that LSTM neural network outperforms ARIMA, Back Propagation NN and various other deep learning models. The major difference that models based on Auto Regressive Moving Average (ARMA) and Auto Regressive Integrated Moving Average (ARIMA) have is that these models focus on linear temporal modelling of time-series data. RNNs and LSTMs on the other hand are essentially nonlinear time series models, where the nonlinearity is learned from the data.

Most of the work in the domain of metro passenger count prediction have primarily taken into account only the historical data from metro stations in the form of smart card data, Automated Fare Collection systems etc. Although there have been initiatives to do prediction using the LSTM network, there has not been sufficient emphasis given to other external parameters such as the meteorological data over that time and the holiday information to explain the occasional anomalies. Also, these predictions do not take into consideration other relevant parameters like gender and age category of people accessing the metro. This has a significant impact on the prediction because the time of the day, the particular day etc affect the different gender and age category of people using the metro. This paper focuses on predicting the passenger flow using the LSTM model with historical data consisting of age and gender statistics, meteorological data and the holiday information.

3. Materials and Methods

For this work, metro data from Kochi Metro Railway Limited, India is selected as case study. Dataset covers hourly information about 333 days, from 15th July 2020 to 30th July 2020. Useful information extracted from each record consists of time stamp, count of passengers, their age and gender. This gender and age information is then mapped into 10 groups, where each group represents a particular age and gender category. For instance, Group 1 represents age category 1 to 15 years having gender male while Group 2 represents age category 1 to 15 years having gender female. Similarly Group 3 represents age category 16 to 30 years having gender male while Group 4 represents age category 16 to 30 years having gender female. Similar pattern of categorization extends for the remaining groups where the age bars are 31-50 years, 51-70 years and lastly age 70 and above.

Another parameter considered is meteorological data over that time period consisting of parameters such as temperature, humidity, pressure etc. In order to explain unusual spikes in passenger flow, additional information

like holiday status of day, local events etc. is also considered. Thus, the input given to the LSTM model consists of 13 parameters initially. The date, time, passenger count, location of the station, five weather parameters-namely a summary of the weather condition, an icon to represent it, temperature, humidity and pressure. Other parameters include group number, day number and whether the day is a holiday or not.

3.1. Long-Short Term Memory Neural Network

LSTM is a modified version of recurrent neural networks. When given the time lags of unknown duration, it can effectively classify, process and predict time series data. The core concept of LSTM is the cell state, which carries relevant information throughout the processing of the sequence. Cell states are updated by adding or deleting information through gates which are basically neural networks. During the training phase these gates learn by maintaining relevant information and discarding the rest. A loop with a few tensor operations forms the control flow of an LSTM network, with predictions done with the help of hidden states.

LSTM performs sequential processing, first it concatenates current input with previous hidden state and load it to the forget layer. Then a candidate layer is created by removing irrelevant data and holding possible values to be added to the new cell state. After computing the forget layer, candidate layer, and the input layer, using the previous cell state the new state is calculated which in turn helps in output calculation. New hidden state is found through point wise multiplication of output with this new cell state.

4.Method

The input variables are normalized to a scale of 0 to 1 before it is given to the LSTM model. Since the parameters that have data in textual form cannot be given as input to the model, it is encoded as integers. The time series data is framed as a supervised learning problem by creating copies of these data columns that are pushed forward for lag observations and similarly, copies of data columns that are pulled backward for forecast observations. This reframed data is split into train and test sets to fit the model. These sets are further split up as input and output variables. Here the count of passengers is taken as the output variable. The input that was in 2D format is reshaped into 3D format [samples, timesteps, features] as expected by the LSTM. An architecture diagram of the proposed model is given in Fig. 1.

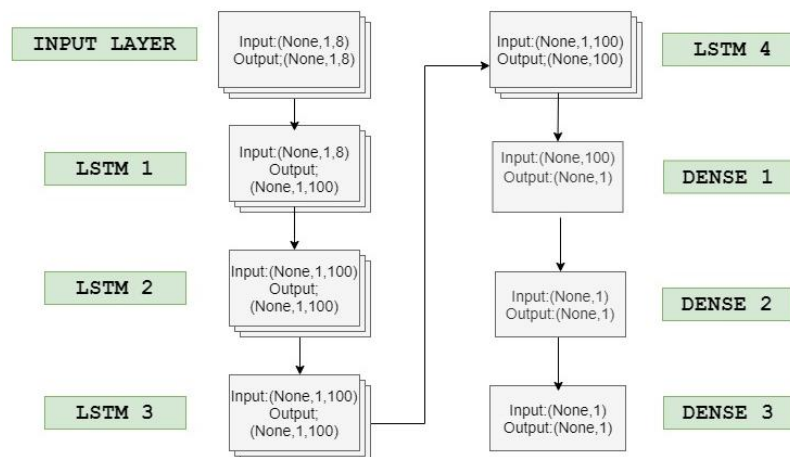


Fig.1. Architecture diagram of the LSTM model

The model learns using Mean Absolute Error (MAE) loss function and Adam optimizer. Unlike the traditional Stochastic Gradient Descent, learning rate for various network parameter weights were separately identified using an Adam optimizer. After this the Root Mean Squared Error (RMSE) is used to calculate the difference between the actual and the predicted values. A detailed study was conducted by training the model with different parameters and the prediction results were thoroughly analyzed. The evaluation is done on the basis of its accuracy and various errors such as the Mean Square Error (MSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE) as given in Eqn. 1 to Eqn. 4. MSE measures the average squared difference between the predicted and actual values. MAE is a linear score which measures the average magnitude of errors in the prediction where the direction is not considered. MAPE is a loss function measured by finding the difference between the average absolute percent error during each time period and the actual value divided by the actual value. The RMSE which is a commonly used metric should be ideally larger than or equal to the MAE. As the difference between these two errors increases, the variance in the individual errors within the sample also increases accordingly.

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

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \tag{2}$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i} 100\% \tag{3}$$

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(\hat{y}_i - y_i)^2}{n}} \tag{4}$$

5. Results and Discussion

5.1 Impact of Features Chosen

Initially all the parameters mentioned earlier were used as input to train the model. But irrelevant or partially relevant features heavily impact the accuracy of the model. Hence, it is necessary to choose the best features to reduce computational costs and ensure good performance of the model. This is done by repeatedly training and evaluating the model by eliminating some of the features. Table 1 given below summarizes the observations. The results show that eliminating the parameter humidity results in an improvement in the accuracy of the model whilst also reducing the overall error. Hence, from this point onwards, we drop the parameter humidity among the features affecting the passenger count. A plot of the various errors as a function of the number of epochs on dropping the parameter humidity is indicated in Fig. 2.

Table 1: Effect of parameters in model accuracy

Eliminated Feature	None	Temperature	Humidity	Pressure	Temperature, Humidity, Pressure	All the Weather Parameters
MSE	4.47%	3.17%	3.65%	4.33%	5.61%	5.75%
MAE	14.69%	11.01%	12.78%	13.22%	17.37%	17.79%
MAPE	93796843.75%	143457737.50%	132147512.50%	188029775.00%	135970437.50%	195812900.00%
Accuracy	0.80%	0.70%	0.90%	0.70%	0.80%	0.70%

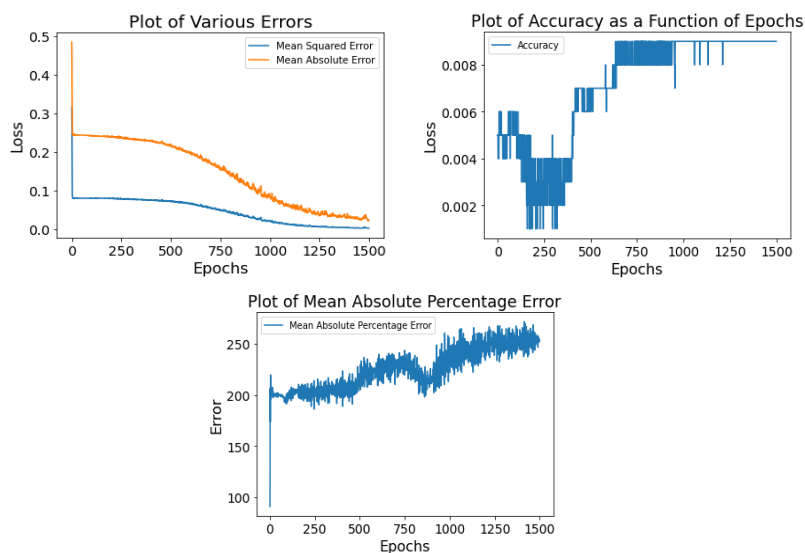


Fig. 2. Plot of various parameters as a function of number of epochs (a) Combined plot of MSE and MAE (b) Accuracy (c) MAPE

5.2 Impact of Dropout Layers

In order to counter the overfitting of the model observed in the training and the validation curves, dropout layers were introduced into the model. Dropout is a technique used for regularization where certain randomly selected neurons are ignored or dropped. This reduces excessive dependence of the model on an individual parameter. It makes the network less sensitive to individual neuron weights and leads to a more generalized network. It learns through multiple independent internal representations. In this paper, different positions of placing the dropout layer were experimented with respect to the LSTM and Dense layers. Best results were obtained on placing the dropout layer after the fully connected Dense layer. Fig. 3 shows the architecture of the model and the placement of dropout layers.

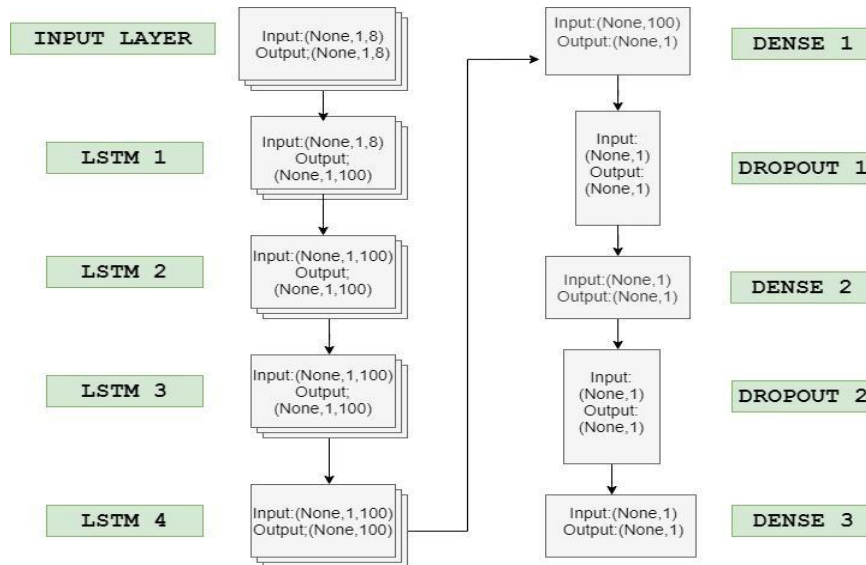


Fig. 3. Architecture diagram of the LSTM model with dropout layers

It was found that the dropout layers led to better performance when used on a larger network with a greater number of hidden layers as this enabled it to learn using more internal representations. Also, different values of dropout probability percentage were experimented with. The error values, accuracy and curves for training and validation loss for each of these cases were examined and the observations are summarized in Table 2 and Fig. 4. It was found that lower dropout rates led to better performance and accurate predictions. As the dropout rates decreased the accuracy of the model improved and the Root Mean Square Error also decreased. For higher values of dropout rates even though the training and validation loss curves converge, accuracy was extremely less. Also, the curves for training and validation loss were found to converge at lower dropout rates. This indicates the reduction in overfitting in the model.

Table 2: Effect of dropout rate in model accuracy

Dronout	Accnracy	RMSE
0.5	0.4%	24.33
0.3	0.2%	23.54
0.2	0.9%	17.53
0.1	0.9%	8.03
0.05	0.9%	2.72
0.01	0.9%	1.89

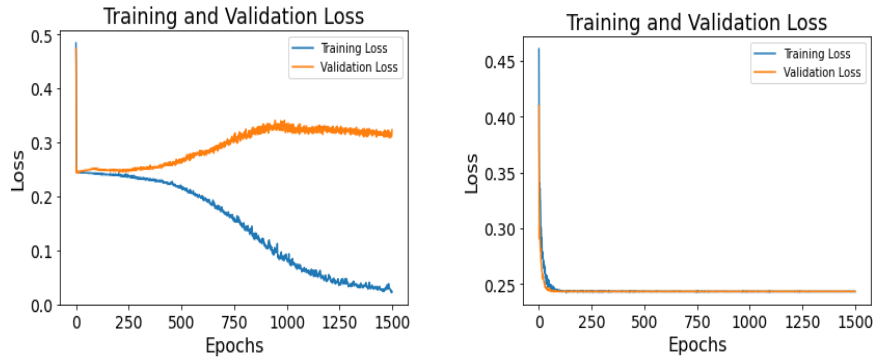


Fig. 4. Plot showing training and validation loss before and after using dropout (dropout rate=0.3%)

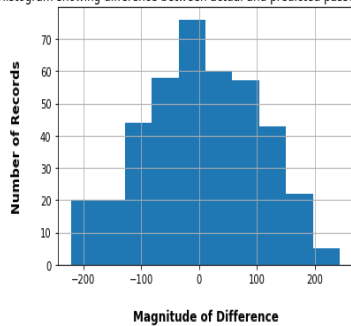
5.3.Effect of Time Lag

The time lag or timestep given to the time series model is an important parameter that affects prediction. It shows the correlation and dependence of current data on data from the previous timestamp or past observations. It is the number of lagged observations of each input variable being used to determine the output variable. We examine the impact of additional contexts of the observation on the prediction. Different values of timestamp were experimented, and the results were summarized in Table 3. Fig.5 indicates the histogram showing the difference between original and predicted passenger count. Here, X-axis represents the difference between the original passenger count and that predicted by the model, while Y-axis represents the number of records. On analysing the results, best result was observed when time lag value is 1. It may be noted that when time lag is 1, peak of the histogram is observed when difference is 0, i.e., when actual and predicted values are matching. This shows that the data has highest dependence on the previous, that is (t-1)th timestep as compared to the t-2, t-3 or t-7th past observation data. Also, unnecessarily high timesteps lead to higher computational cost and time.

Table 3: Effect of Time Lag in model accuracy

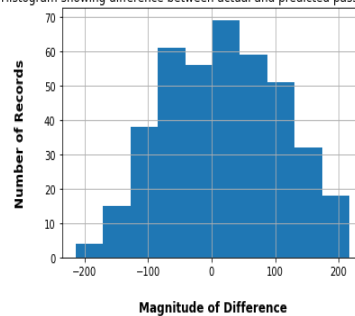
Time	1	3	5	7
Accurac	0.90	0.90	0.80	0.8
MSE	0.07	0.14	0.38	0.01
MAE	1.05	1.89	5.29	0.64
MAPE	1103315	708472	53714287.	1824972.66
RMSE	1.05	1.89	5.29	0.64

Histogram showing difference between actual and predicted passenger count



(a)

Histogram showing difference between actual and predicted passenger count



(b)

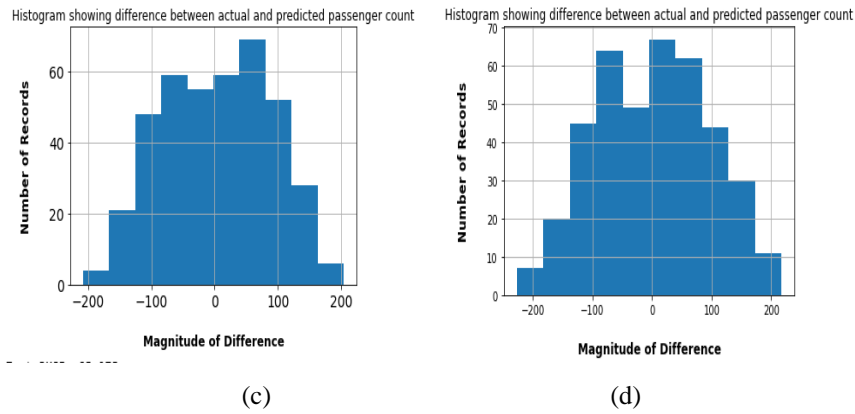


Fig. 5. Plots of histogram indicating the difference between the original and predicted values. (a), (b), (c) and (d) represents cases for time lag values 1, 3, 5 and 7 respectively.

5.4. Effect of Learning Rate

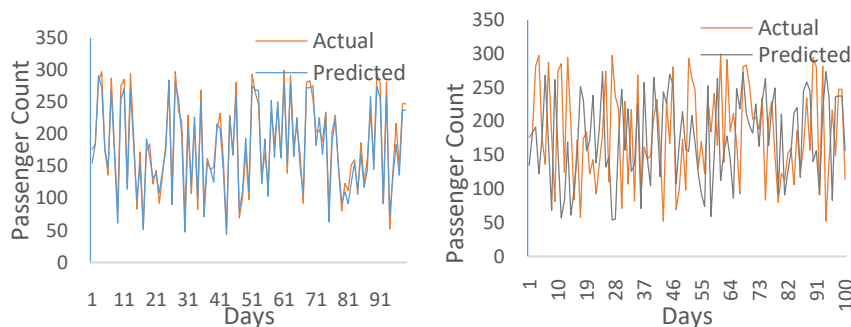
The learning rate is a parameter which takes values between 0 and 1 and determines the amount by which the weights are updated in the model. Different values of learning rate were experimented with and the results are as summarized in Table 4. The best performance is observed for learning rate value 0.001 which is also the default rate for the Adam optimizer used by the model. The accuracy is highest and RMSE and overall errors are minimum at this rate. Even though higher values of learning rate than the ones indicated in the table were experimented with, they gave abnormal results.

Table 4. Effect of learning rate in model accuracy

Learning	0.00000	0.00001	0.001	0.01	0.1
Accuracy	0.50	0.50	0.80	0.50	0.50
MSE	28.84	8.15	1.48	7.99	8.34
MAE	45.85	24.50	8.40	24.33	24.76
MAPE	200460	229679900.0	112642237.5	24917770	218928500
RMSE	45.85	24.50	8.40	24.33	24.76

5.5. Comparative Analysis

It is difficult to use the traditional regression approaches when the underlying dependence between the variables is unknown or does not have a predefined definition. Also, when there is no stationarity or proper trend in the data. In this paper, we compare the model against the traditional K Nearest Neighbour Algorithm. We test the accuracy of the model for various K values. It was seen that the RMSE error for KNN is larger than the error obtained for the LSTM model for the different K values. KNN being a lazy learner, stores all the training data and only when it has to make a prediction, does it search for the nearest neighbour or similarity in features to stored data to make the prediction.



(a) (b)

Fig. 6. Predicted Passenger Count for various days (a),(b) represents LSTM Model and KNN Model respectively

The LSTM model on the other hand, accurately models the time connections to the past data. It was also found that irrespective of the amount of similarity to the past data or the correlation between the various features, the LSTM model could train using the previous data, to find the complex underlying relationships and make quality predictions. Fig. 6 shows the comparison between the two models for the first 100 predictions for ease of observation and legibility. It is clear from the figure that the predictions of the LSTM model outperforms KNN. Thus, we validate the LSTM model as a useful alternative against the traditional approaches for this problem.

From the above experiments, the impact of the various parameters on model training and subsequent predictions were understood. It was found that the feature humidity negatively impacted the performance of the model. Thus, humidity was not considered as a parameter when determining the passenger crowd in the metro. Introducing dropout layers led to reduction in overfitting of the model. Different configurations of dropout were experimented with, and it worked best when used with a larger network and for lower dropout rates. Also, it was found that the best results were obtained for timestep value 1. The current count was found to be dependent the most on the count at $(t-1)^{th}$ timestep than the timesteps before it. Finally, the impact of learning rate was evaluated, and it yielded best results for value 0.001. Both values higher and lower than this did not lead to much improvement in the training.

With all the parameters configured as discussed above, the accuracy of the prediction was found to improve significantly. The overall error rates also decreased. Thus, the passenger count was predicted on an hourly basis using the LSTM model. The results obtained from prediction, can further be analysed to get an overall crowd pattern in the metro station. For instance, the passenger count predicted can be grouped and analysed on the basis of weekends, weekdays and holidays. This can benefit staff and resource allocation by allotting more staff, ticket and security check counters on the peak days and at rush hours. In addition, the gender and age wise statistics predicted can be used for playing targeted advertisements for the specific age categories which are likely to come at that hour and for other commercial purposes. In short, the parameters chosen to determine the passenger count themselves can be used for finding useful trends and patterns based on the amount of passenger crowd expected. This will help to ensure passenger safety and an overall smooth functioning of the metro station.

6. Conclusion

In this paper, we predict the count of passengers to be expected in the metro station on an hourly basis using the LSTM neural network. Unlike previous works in this area, we take into consideration external factors like the meteorological conditions and the holiday information of that particular day. Emphasis is also given to the gender and age category to which these passengers belong. Different configurations of the LSTM network were explored in terms of the parameters used for prediction, the dropout layers and dropout rate, the dependence of the model on the previous time step and the impact of learning rate on the model. The configuration that gives the best result for prediction were analysed. The efficiency of the model was also evaluated against traditional models such as the KNN and was found to yield better results.

7. Acknowledgements

This work is supported by Department of Science and Technology, Government of India (Project No: DST/ICPS/CPS-Individual/2018/1091) and Kochi Metro Rail Limited, India

References

- Jiayu Qiu, Bin Wang and Changjun Zhou, "Forecasting stock prices with long-short term memory neural network based on attention mechanism", PLOS ONE 15, January, 2020.
- S. Poornima and M. Pushpalatha, "Prediction of Rainfall Using Intensified LSTM Based Recurrent Neural Network with Weighted Linear Units", Atmosphere, Vol. 10, Issue 11, 2019.
- YiFei Li and Han Cao, "Prediction for Tourism Flow based on LSTM Neural Network", Proceedings of International Conference on Identification, Information and Knowledge in the Internet of Things Prediction, 2017, Procedia Computer Science, Volume 129, 2018.
- Halit Apaydin, Hajar Feizi, Mohammad Taghi Sattari, Muslume Sevba Colak, Shahaboddin Shamshirband and Kwok-Wing Chau, "Comparative Analysis of Recurrent Neural Network Architectures for Reservoir Inflow Forecasting", Water, Volume 12, Issue 5, 2020.
- Ali Zonoozi, Jung-jae Kim, Xiaoli Li and Gao Cong, "Periodic-CRN: A Convolutional Recurrent Model for Crowd Density Prediction with Recurring Periodic Patterns", Proceedings of the Twenty Seventh International Joint Conference on Artificial Intelligence, 2018.

- Yirong Zhou , Hao Chen, Jun Li , Ye Wu, Jiangjiang Wu and Luo Chen, “Large-Scale Station-Level Crowd Flow Forecast with ST-Unet”, *International Journal of Geo-Information*, Vol. 8, Issue 3, 2019.
- Yong Han, Shukang Wang, Yibin Ren, Cheng Wang, Peng Gao and Ge Chen, “Predicting Station-Level Short-Term Passenger Flow in a Citywide Metro Network Using Spatiotemporal Graph Convolutional Neural Networks”, *International Journal of Geo-Information*, Vol. 8, Issue 6, 2019.
- Ling X, Huang Z, Wang C, Zhang F, Wang P (2018), “Predicting subway passenger flows under different traffic conditions”, *PLOS ONE* 13, August, 2018.
- Xiaoqing Dai , Lijun Sun and Yanyan Xu, “Short-Term Origin-Destination Based Metro Flow Prediction with Probabilistic Model Selection Approach”, *Journal of Advanced Transportation*, Volume 2018. Article ID 5942763.
- Zhu Hai-yan, “N Days Average Volume Based ARIMA Forecasting Model for Shanghai Metro Passenger Flow”, *IEEE Proceedings of International Conference on Artificial Intelligence and Education (ICAIE)*, Hangzhou, China, 2010.
- Danfeng Yan, Junwen Zhou, Yao Zhao and Bin Wu , “Short-Term Subway Passenger Flow Prediction Based on ARIMA”, *Proceedings of International Conference on Geo-Spatial Knowledge and Intelligence*, 2017, *Journal of Geo-Spatial Knowledge and Intelligence*, Springer, Singapore, 2018.
- Soheila Mehrmolaei and Mohammad Reza Keyvanpour , “Time series forecasting using improved ARIMA”, *Proceedings of Artificial Intelligence and Robotics (IRANOPEN)*, Qazvin, Iran, 2016, IEEE.