LSTM Views Forecast Model

Task: Predict the number of views of TikTok's Biggest influencers for the next 30 days so that they can plan for cloud computing costs. This information is vital for TikTok to know before their systems crash.

Methodology of Approach

To predict the number of views that TikTok's biggest influencers will get, an LSTM Model was used. To conduct this, information was collected on TikTok Influencers Data. This included the day the data was collected, the TikToker's ID , the Category of the video being posted, the number of views, the number of Likes and the number of Followers that the TikToker had at the time of data collection.

The methods that were taken for preparing the model for the machine learning process included importing the necessary files and packages for the model in a juypter notebook, loading the csv file to a pandas data frame and conducting data cleaning procedures such as removing leading and trailing whitespaces from all the column, removing commas from the views column, and dropping the TikTokerID and Category column from the data frame and storing the changes into a new dataframe called new_df.

Preparing the model for the Neural Network

We summed the views for each day (Day 1, Day 2 etc) in the dataset and then we normalized this data to make it follow a stationary pattern. This is done so that the neural network model could effectively handle values with little to no variance. Also, we conducted a train/ test split where 20% of the training data was reserved for the test set. A sequence length of 10 was also specified so that the model can use the last 10 views values in the time series to predict the next aggregated views value. This is because we have 10 views in each aggregated value, and we can use all these values in total to predict the next daily aggregated views value.

Building the Input/ Output Structure of the LSTM Model

A function called create_sequence was created to generate the input-output pairs for training the LSTM model. This involved making X (the independent variable) be a 2D array where each row represents a sequence of 10. As stated before, this sequence value is specified to take the numbers of the last 10 views in the data set. The y value (the dependent variable / predicted variable) is a 1D array that represents the next value in the sequence. The X array captures the last 10 values from the tiktok_influencers_data.csv dataset, and the y array predicts the subsequent value based on this sequence. The objective is to train the model to predict the next views value using the preceding 10 views. The X and y arrays were split into training and testing sets in an 80/20 ratio, as described earlier.

Training the LSTM Model

This step encompassed defining, compiling, and training an LSTM model, incorporating dropout regularization techniques, and employing early stopping to mitigate the effects of model overfitting.

The LSTM model is defined with two LSTM layers of 50 units each. The first LSTM layer returns the views sequences, and the second LSTM layer returns the output of the last timestep. This means that for each of the number of aggregated daily views in the model, the model is estimating what the next likely aggregated view will be based on the past 10 aggregated number of views. Each of these layers are followed by a dropout layer and an Output layer that returns the predicted aggregated y_pred aggregated views value for each day.

There were concerns for overfitting given the limited diversity and noise sensitivity of our dataset (500 entries) especially after overfitting was noticed after training our model on the first few occasions. To combat this, dropout regularization and early stopping were employed, and this enhanced the robustness of the model, improving its ability to generalize effectively to new data. This was particularly observed when we applied a dropout rate of 40% to the dropout layers in the model's neural network weight update cycle.

The model was trained on these hyperparameters: Number of epochs = 50, the learning rate = 0.001, the batch size was 8, the dropout rate = 0.4 as previously stated, and a verbose value of 1 was chosen. The results of the Model loss were plotted on a graph, and it was observed that there was a noticeable decrease in the trend which highlights that the model has learnt the training data well.

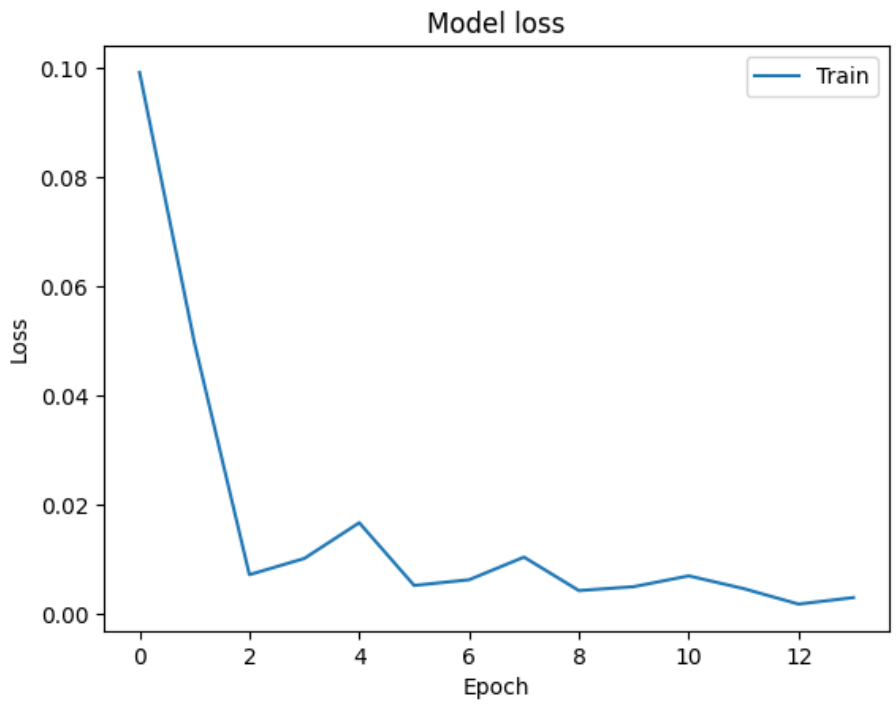

Image of the Model Loss which signifies that the model learnt the training data patterns based on its downward trend.

After training the model, we evaluated the trained model on the test data using the mean squared error loss function. The model obtained a loss of 0.00024130351 which means that on average, the squared difference between the predicted and true values on the test set is 0.00024130351.

Forecasting the Future Views

In the end, we calculated the *Mean Squared Average (MAE), Mean Squared Error (MSE) and Root Mean Squared Error (RMSE)* for both the train and test set.

These were the values obtained.

Metric	Train	Test
MAE	1.47117e+07	1.93626e+06
MSE	2.35782e+14	5.09571e+12
RMSE	1.53552e+07	2.25737e+06

Based on these values, our model performed better on the test set for all the metrics: MAE, MSE and RMSE indicating that our model had generalized well to the test set.

The Model's forecast for the total number of views in the next 30 days is 271,928,960 views across all the categories mentioned in the dataset.

Advice: Increase the Cloud computing coverage for their server systems to avoid system crashes.

Assumptions and Limitations

- **Assumption of Stationarity:**

The normalization process assumes stationarity in the data, which might not hold true if there are underlying trends or seasonality in the views data that are not considered in the model.

- **Simplified Input Features:**

The model relies solely on historical views data without incorporating additional features such as likes, followers, or categorical information about video content. Incorporating more features could enhance the model's predictive capabilities.

- **Generalization Challenges:**

Generalizing the model to predict the views for all TikTok's biggest influencers might be challenging, as the dataset may not fully represent the diversity and dynamics of all influencers on the platform.

- **Forecasting Horizon Limitation:**

The model is designed to predict views for the next 30 days, but the accuracy of long-term forecasts can be challenging, and the model's performance might vary for different forecasting horizons.

- **Model Interpretability:**

The LSTM model is a complex architecture, and interpreting the inner workings of the model may pose challenges, limiting the ability to understand how specific features contribute to predictions. This is because it is a black box model.