

## **COMPARATIVE ANALYSIS OF THE THREE FORECASTING MODELS**

In this project, we have plotted below requirements using the three forecasting methods LSTM, Prophet and Statsmodel. Let us evaluate their performance based on data pre-processing, model-training, accuracy, and visualizations. We will then see the factors on how to choose the right model.

Forecasted data for:

1. The day of the week maximum number of issues created
2. The day of the week maximum number of issues closed
3. The month of the year that has maximum number of issues closed
4. Plot the created issues forecast
5. Plot the closed issues forecast
6. Plot the pulls forecast
7. Plot the commits forecast
8. Plot the branches forecast
9. Plot the contributors forecast
10. Plot the release forecast.

### **DATA PREPROCESSING:**

TensorFlow/Keras LSTM:

- Requires careful normalization and scaling of input features.
- Handles sequential data well, making it suitable for time series forecasting.

Prophet:

- Handles missing data and outliers well.
- Automatically detects seasonality and holidays, reducing the need for extensive feature engineering.

Statsmodel:

- May require more manual feature engineering and preprocessing.
- Sensitive to missing values, requiring careful handling.

## **MODEL TRAINING:**

TensorFlow/Keras LSTM:

- This model needs extensive training. We considered 80% train data and 20% test data for the experiments.
- Hyperparameter tuning may be required for optimal performance

Prophet:

- Quick to set up with fewer hyperparameters to tune.
- Designed to handle daily observations with seasonality.

Statsmodel:

- Choice of statistical model depends on the time series characteristics.
- Training involves estimating model parameters, with potential challenges in convergence.

## **ACCURACY:**

TensorFlow/Keras LSTM:

- May require significant computational resources, especially for large networks as these graphs took the longest time.

Prophet:

- Generally more computationally efficient than deep learning models.
- Automatically detects seasonality and holidays, reducing the need for extensive feature engineering.

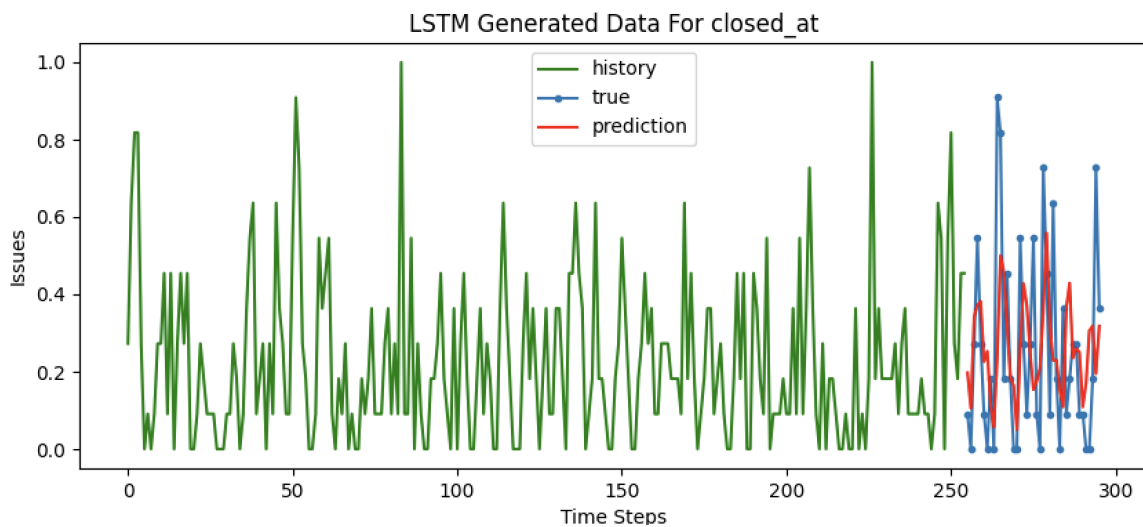
Statsmodel:

- May require more manual feature engineering and preprocessing.
- Sensitive to missing values, requiring careful handling.

## VISUALIZATIONS:

### TensorFlow/Keras LSTM:

- Can generate detailed visualizations of model architecture as shown below.
- Visualizing internal mechanisms of neural networks can be complex.

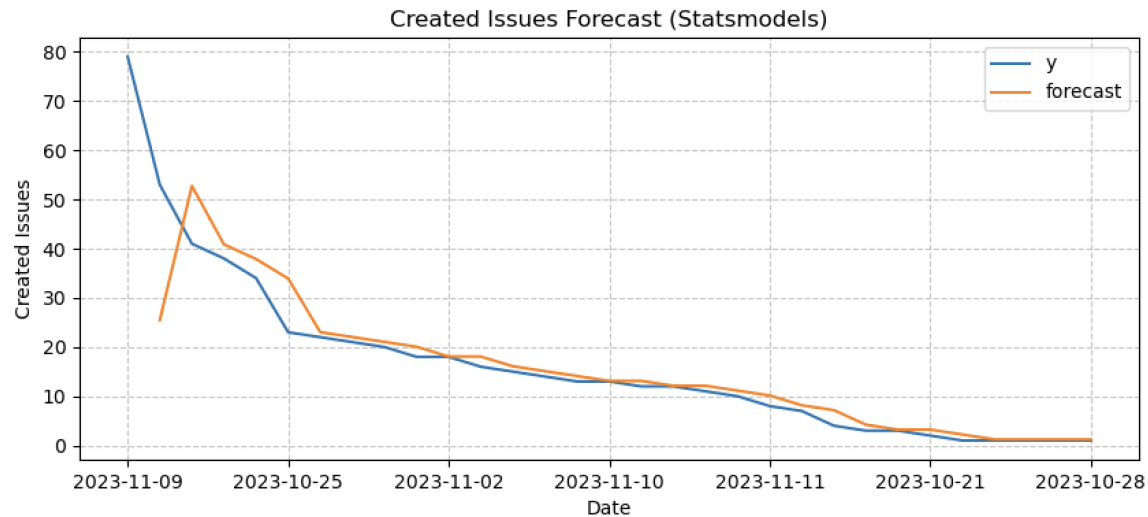


### Prophet:

- Built-in visualization tools for inspecting trends, seasonality, and holidays.
- Offers intuitive plots for both historical data and forecast.
- Limited customization options for visualizations compared to more flexible libraries.

### Statsmodel:

- Offers standard statistical model diagnostics like residual plots as shown below.
- Users can customize visualizations based on statistical model output.
- Visualizations may not be as intuitive or automatic as in other frameworks.



## HOW TO CHOOSE THE RIGHT MODEL?

**Complex Patterns** -> Choose TensorFlow/Keras LSTM as Statsmodel and Prophet are limited to capturing simpler patterns.

**Large Datasets** -> For large datasets choose TensorFlow/Keras LSTM or Prophet as Statsmodel may not scale well for large datasets

**Ease of use** -> If we want to choose a user-friendly model, go for Prophet or Statsmodel as LSTM needs deep learning experience.

**Seasonality and Holidays Predictions** -> Choose Prophet as it has in-built capabilities for handling seasonality and holidays.

**Computational Efficiency** -> If we want to save on computational efficiency, go for Prophet or Statsmodel as LSTM is resource-intensive.

**Forecasting Horizon** -> If we want to make long-term predictions choose LSTM or Prophet as Statsmodel performance may vary for long term predictions.

## Summary:

TensorFlow/Keras LSTM:

- Moderate ease of use, especially for users with deep learning experience.
- Provides detailed but potentially complex visualizations.

Prophet:

- Highly user-friendly, especially for users without extensive technical expertise.
- Offers clear and intuitive visualizations.

Statsmodels:

- User-friendly for those with a statistical background.
- Visualizations may require more manual customization.