

Case Study

AI-Generated Predictive Analysis

www.tekbank.net

Background

Delay in train arrivals is a common phenomenon and it needs to be efficiently managed to minimize any time delay for passengers waiting to commute in a train. The primary objective of this project was to analyze the existing delay in train schedules and later predict the time delay for any given time and train routes, considering various factors that influence the arrival of a train to the station. Our client provides public train commute services and manages scheduling and operation of trains on various train routes. The delay in schedules for trains on routes they are operated has to be managed with the least possible and predictable delay time.

Maintaining a predictable train delay impacts the commute pattern of passengers and impacts the frequency of train services on all routes.

TekBank used the following technologies in developing our Predictive Analytics solution:

- ⚙️ **Data Engineering using Time series and meta data**
- ⚙️ **Machine Learning Operations (MLOPS)**
- ⚙️ **Deep AR on Amazon Web service**
- ⚙️ **Professional Visualization using Tableau**
- ⚙️ **Test and train ML models using DeepDR+ Algorithms**
- ⚙️ **Amazon S3 captures training dataset**

Approach

Data Extraction

The ingestion of huge volumes of train delay time data in real time was needed to understand the existing pattern of delay time. Data ingestion are delay data, route data, and schedule data. The data ingested was split and used for training and testing models which were developed after MLOPS performing data digestion and information extraction.

AI-Generated Predictive Analysis

Approach (Continued)

Data Analysis

The data analysis is the crucial part of understanding how the train delay time happens in real-time and it helps in identifying key parameters that shape the train delay time data set. The analysis starts with target data set which was uploaded from AWS Glue data integration platform into the Amazon S3 bucket. This ingested target data was then cleansed for basic sanity and was transformed into a valid dataset. Amazon step functions or Lambda functions processed basic data wrangling operations and made the data clean for further machine learning processes.

Training and Testing

The training set was also extracted from the target data set at the data ingestion phase and used to train the model developed using machine learning algorithms. The Amazon Glue data integration platform ingests target data in CSV format into Amazon S3 as training data along with transformed data into ML Lambda function. This is a part of MLOPS that was automated to learn subsequently in real-time.

Predicting Delay Time

The training data set extracted from the target data set was fed into the Amazon step function or AWS Lambda. Amazon Forecast used to perform supervised machine learning techniques which use time series and hyper parameterization function. Together known as DeepAR+ is a time series based supervised algorithm that is inside Amazon forecast to keep up with the prediction of delay time from real time data ingestion.

Results

The result of this project aims at developing a MLOPS workflow using AWS web services cloud computing platform to store and retrieve data sets in real time. The following is the nature of workflow that was intended to be adhered to achieve a better delay time prediction capability using MLOPS.

- ⚙️ **Creating a target dataset**
- ⚙️ **Ingest data set into Data Integration platform**
- ⚙️ **Import data as training and test data set in the pre-processing stage**
- ⚙️ **Create a predictor model using Deep AR+ algorithm and train the model**
- ⚙️ **Create a forecast on how delay time is expected to be in the future (for the next 14 days)**
- ⚙️ **Update data sources and other model parameters in real-time**
- ⚙️ **Reduce error rate in the forecasted schedule over the actual schedule**

Visualization of the Data – Tableau

The interpretation of the comparison of delay minutes grouped was by delay code between the actual delay time and the predicted delay time. This dashboard is updated every two weeks in production, based on forecast.