1.To improve arrival time prediction accuracy based on historical data and traffic condition:

Traffic prediction has always been a challenge for transportation planners and city managers. With the increasing growth of cities and the number of vehicles on the roads, the need for accurate and reliable traffic predictions has become more pressing. In recent years, machine learning has shown great promise in solving this problem.

Traffic prediction involves estimating the future behavior of traffic in a particular area. This information is useful for a variety of purposes, including **reducing congestion, optimizing transportation systems, and improving road safety**. In the past, traffic prediction has been based on traditional methods such as rule-based models and time-series analysis. However, these methods are often limited in their ability to capture the complexity and variability of traffic patterns.

Machine learning, on the other hand, is well-suited to handle large and complex datasets, making it an ideal tool for traffic prediction. Machine learning algorithms can automatically identify patterns and relationships in traffic data and use these to make predictions about future traffic conditions.

There are several types of machine learning algorithms that can be used for traffic prediction, including **regression, time-series analysis, and artificial neural networks**. Regression models use historical traffic data to predict future traffic conditions based on past trends. Time-series analysis models look at the patterns in traffic data over time and use these patterns to make predictions. Artificial neural networks, which are modeled on the structure of the human brain, are also commonly used for traffic prediction.

One of the key advantages of machine learning for traffic prediction is its ability to handle large and complex datasets. For example, traffic data may include information on traffic flow, vehicle speed, and traffic density, as well as other factors such as weather conditions, road conditions, and time of day. Machine learning algorithms can process this data and identify the most important factors that influence traffic patterns, making them ideal for traffic prediction.

Another advantage of machine learning for traffic prediction is its ability to adapt to changing conditions. Traditional traffic prediction methods are often limited in their ability to handle changes in traffic patterns, but machine learning algorithms can automatically adjust to these changes and continue to make accurate predictions.

Code Implementation

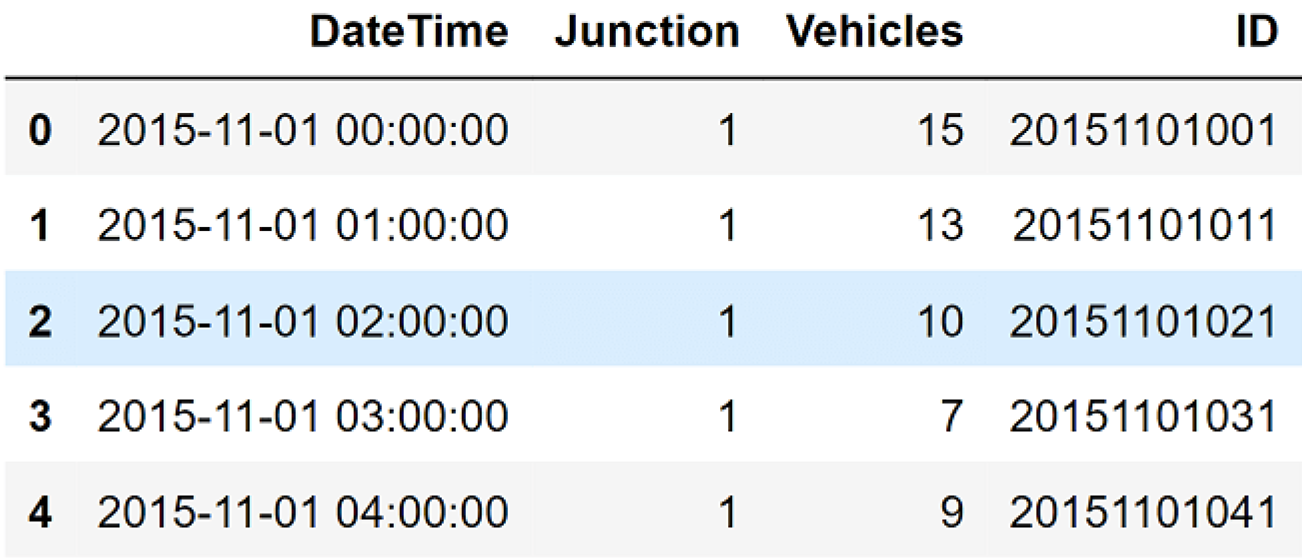
Importing Libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import datetime
import tensorflow
from statsmodels.tsa.stattools import adfuller
from sklearn.preprocessing import MinMaxScaler
from tensorflow import keras
from keras import callbacks
from tensorflow.keras import Sequential
from tensorflow.keras.layers import Conv2D, Flatten, Dense, LSTM, Dropout, GRU, Bidirectional
from tensorflow.keras.optimizers import SGD
import math
from sklearn.metrics import mean_squared_error

import warnings
warnings.filterwarnings("ignore")
```

```
dataset = pd.read_csv("traffic.csv")
dataset.head()
```

Output:



	DateTime	Junction	Vehicles	ID
0	2015-11-01 00:00:00	1	15	20151101001
1	2015-11-01 01:00:00	1	13	20151101011
2	2015-11-01 02:00:00	1	10	20151101021
3	2015-11-01 03:00:00	1	7	20151101031
4	2015-11-01 04:00:00	1	9	20151101041

The traffic data comes from several time periods since the sensors on each of these intersections were gathering data at different times. Data from several of the intersections were scarce or restricted.

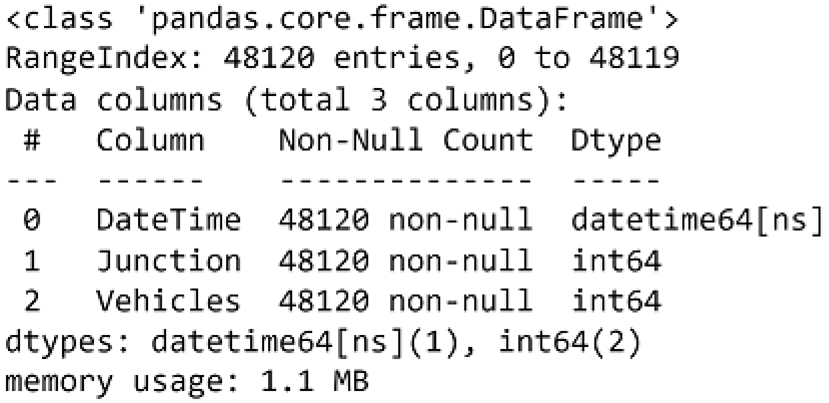
Data Exploration

- Feature engineering for EDA
- Plotting time series
- Parsing dates

2:Traffic condition used in arrival time:

- data frame to be used **for** EDA
- data frame=dataset copy
- # Let's plot the timeseies
- colours = ["#FFD4DB","#BBE7FE","#D3B5E5","#dfe2b6"]
- plt.figure(figsize=(20,4),facecolor="#627D78")

- Time_series=sns.lineplot(x=dataframe['DateTime'],y="Vehicles",data=dataframe, hue="Junction", palette=colors)
- Time_series.set_title("Years of Traffic at Junctions")
- Time_series.set_ylabel("Vehicles in Number")
- Time_series.set_xlabel("Date")
- **Output:**



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48120 entries, 0 to 48119
Data columns (total 3 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   DateTime  48120 non-null  datetime64[ns]
 1   Junction  48120 non-null  int64
 2   Vehicles  48120 non-null  int64
dtypes: datetime64[ns](1), int64(2)
memory usage: 1.1 MB
```