IoT_SmartParking_Feature_Engg

February 25, 2025

0.1 AAI-530 Group 11 - IoT Smart Parking Management - Feature Engineering

[2]:
```python
# import necessary libraries for data I/O and feature engineering
import pandas as pd
import numpy as np
```

[3]:
```python
# Load dataset IIoT Smart Parking Management real-time dataset
df = pd.read_csv("IIoT_Smart_Parking_Management.csv")
```

[4]:
```python
# view the dataframe
df
```

[4]:
```
                         Timestamp  Parking_Spot_ID  Sensor_Reading_Proximity  \
0    2021-01-01 00:00:00.000000000               20                  1.023651
1    2021-01-02 06:39:16.756756756               49                  3.903349
2    2021-01-03 13:18:33.513513513               38                 10.315709
3    2021-01-04 19:57:50.270270270               31                  6.588039
4    2021-01-06 02:37:07.027027027                8                  8.213969
..                          …                …                         …
995  2024-06-24 21:22:52.972972960                5                  5.349471
996  2024-06-26 04:02:09.729729728               47                 15.688164
997  2024-06-27 10:41:26.486486480                7                  0.357255
998  2024-06-28 17:20:43.243243232               49                  0.293735
999  2024-06-30 00:00:00.000000000               23                  1.657731

     Sensor_Reading_Pressure  Vehicle_Type_Weight  Vehicle_Type_Height  \
0                   1.541461          1831.770127             4.392528
1                   1.621719          1330.815754             4.595638
2                   6.292374          1255.134827             4.313721
3                   1.659870          1523.442919             3.567329
4                   3.278467          1758.490837             5.145836
..                    …                    …                    …
995                10.515457          1267.050258             4.442869
996                 2.661805          1547.138376             4.413585
997                 1.411642          1552.856947             4.380228
998                12.630766          1299.945385             4.091230
999                 7.449078          1559.375000             6.684525
```

```
      User_Type  Weather_Temperature  Weather_Precipitation  \
0       Visitor            18.092553                      1
1    Registered            13.397533                      0
2    Registered            21.687410                      0
3       Visitor            18.683461                      0
4       Visitor            19.214876                      0
..          …                  …                    …
995     Visitor            19.430937                      0
996     Visitor            25.426111                      0
997  Registered            20.192776                      0
998  Registered            17.581707                      0
999  Registered            18.766378                      0

    Nearby_Traffic_Level  …  Occupancy_Status  Vehicle_Type  \
0                    Low  …          Occupied           Car
1                    Low  …          Occupied           Car
2                   High  …            Vacant           Car
3                 Medium  …            Vacant    Motorcycle
4                   High  …          Occupied           Car
..                   …  …               …           …
995                  Low  …            Vacant           Car
996                  Low  …            Vacant           Car
997                  Low  …            Vacant           Car
998               Medium  …          Occupied           Car
999               Medium  …          Occupied           Car

     Parking_Violation  Sensor_Reading_Ultrasonic  Parking_Duration  \
0                    0                 102.951052                 4
1                    0                  87.559131                 3
2                    1                 100.061854                 5
3                    1                 110.594598                 2
4                    0                  84.786963                 2
..                 …                        …               …
995                  0                 105.332652                 2
996                  1                 124.841337                 2
997                  0                  93.011015                 1
998                  1                  89.972326                 2
999                  0                  97.877279                 2

     Environmental_Noise_Level Dynamic_Pricing_Factor  Spot_Size  \
0                    55.620740                    0.8   Standard
1                    56.682386                    1.2    Compact
2                    59.239322                    0.8   Standard
3                    44.545155                    0.8   Standard
4                    48.012604                    0.8   Standard
..                         …                    …        …
995                  69.507857                    0.8  Oversized
```

```
996                 49.958346                    1.5   Standard
997                 60.676107                    1.0   Standard
998                 56.465611                    1.2  Oversized
999                 44.105778                    1.2   Standard

    Proximity_To_Exit User_Parking_History
0            6.610474             6.660310
1            8.678719             6.766187
2           13.795262            -0.910052
3            1.678721            10.415888
4           20.012252             4.355544
..                …                    …
995          3.686763             1.749779
996         11.989485             2.569270
997          4.265255            11.013160
998          5.713190             4.561407
999          2.691136             8.600266

[1000 rows x 28 columns]
```

[6]:

```python
# import necessary python libraries
import pandas as pd
import numpy as np
# for data pre-processing, MinMaxScaler, One Hot Encoding and Label Encoding
from sklearn.preprocessing import MinMaxScaler, OneHotEncoder, LabelEncoder

# Load dataset
df = pd.read_csv('IIoT_Smart_Parking_Management.csv')

# Ensure 'Timestamp' is in datetime format
df['Timestamp'] = pd.to_datetime(df['Timestamp'])

# Time-based features
df['Hour'] = df['Timestamp'].dt.hour
df['DayOfWeek'] = df['Timestamp'].dt.dayofweek
df['Month'] = df['Timestamp'].dt.month
df['IsWeekend'] = (df['DayOfWeek'] >= 5).astype(int)

# Convert 'Occupancy_Status' to numeric before calculating rolling average
# Assuming 'Occupied' maps to 1 and 'Vacant' maps to 0
df['Occupancy_Status_Numeric'] = df['Occupancy_Status'].map({'Occupied': 1,␣
 ↪'Vacant': 0})

# Rolling averages
df['RollingAvg_Occupancy'] = df['Occupancy_Status_Numeric'].rolling(window=5,␣
 ↪min_periods=1).mean()
```

```
# Lag features
df['Prev_Occupancy'] = df['Occupancy_Status_Numeric'].shift(1)
df['Prev2_Occupancy'] = df['Occupancy_Status_Numeric'].shift(2)
df.fillna(0, inplace=True)  # Fill NaN values caused by shifting

# Simulated weather data (replace with real data if available)
df['Rainfall'] = np.random.choice([0, 1], size=len(df))  # 0 = No Rain, 1 = Rain
df['Temperature'] = np.random.randint(15, 35, size=len(df))  # Simulated␣
 ↪temperature values

# Aggregated features
df['Hourly_Occupancy'] = df.groupby('Hour')['Occupancy_Status_Numeric'].
 ↪transform('mean')
df['Daily_Occupancy'] = df.groupby('DayOfWeek')['Occupancy_Status_Numeric'].
 ↪transform('mean')

# One-Hot Encoding for categorical features
encoder = OneHotEncoder(sparse_output=False)
encoded_features = encoder.fit_transform(df[['DayOfWeek']])
df = pd.concat([df, pd.DataFrame(encoded_features, columns=encoder.
 ↪get_feature_names_out())], axis=1)
df.drop(columns=['DayOfWeek'], inplace=True)

# Normalize numerical features
scaler = MinMaxScaler()
numeric_columns = ['RollingAvg_Occupancy', 'Prev_Occupancy', 'Prev2_Occupancy',␣
 ↪'Hourly_Occupancy', 'Daily_Occupancy', 'Temperature']
df[numeric_columns] = scaler.fit_transform(df[numeric_columns])

# Save processed dataset
df.to_csv('IoT_SmartParking_Processed.csv', index=False)

print("Feature engineering completed and saved as 'IoT_SmartParking_Processed.
 ↪csv'")
```

```
Feature engineering completed and saved as 'IoT_SmartParking_Processed.csv'
```

The exported feature engineered IoT_SmartParking_Processed.csv file is used by LSTM, XGBoost and RNN for Occupancy Status of Smart Parking Slot