LOTUS-PARK: Intelligent Real-Time Parking Assistance using Machine Learning in Urban Smart Mobility Environments

Kandimalla Sai Ruthik[1], Lalam Pranay[2], Kalamalla Naveen Kumar Reddy[3], Mangali Veerendra[4] and K. Lakshmi[5]
{ksairuthik2004@gmail.com[1], lalampranay@gmail.com[2], k.naveen1564@gmail.com[3], veeruveerendra71@gmail.com[4] ,lakshmicse@gpcet.ac.in[5]}

Final year, Department of Computer Science and Engineering, G. Pullaiah College of Engineering and Technology (Autonomous), Kurnool, Andhra Pradesh, India[1, 2, 3, 4]
Assistant Professor, Department of Computer Science and Engineering, G. Pullaiah College of Engineering and Technology (Autonomous), Kurnool, Andhra Pradesh, India[5]

Abstract. LOTUS-PARK, a novel metadata-driven machine learning framework for intelligent, real-time parking assistance in urban environments. Unlike most traditional systems that process images, LOTUS-PARK employs structured contextual data (temporal, spatial, environmental and vehicle related features) to derive predictions of parking occupancy and suggest the optimal parking locations. The ability of the proposed methodology to do multi-stage learning is evidence of its abilities by capturing both pattern recognition and temporal stability, in addition to its incorporation of a predictive GreenScore metric to model sustainability. Experiments performed on a custom generated urban parking dataset show that LOTUS-PARK outperforms baseline models (Logistic Regression and Random Forest) with 0.90 of accuracy, 0.88 of F1-score with a precision of 0.91. Moreover, the system has a strong capability of predicting eco-efficiency, as the GreenScore regression R^2 is 0.96 with an MSE of 0.002. Analysis of occupancy trends across variables such as days, hours, weather, and lighting conditions confirm the model's robustness and flexibility. LOTUS-PARK offers a sustainable and intelligent solution for optimizing smart mobility and urban parking with high predictive accuracy.

Keywords: Intelligent Parking Systems; Machine Learning; Smart Mobility; Urban Parking Optimization; Sustainable Transportation

1 Introduction

Urbanization is very rapid and as a result the vehicle density is increasing and more and more pressure on the transportation infrastructure is growing in the metropolitan cities [1,2]. Drivers lost in search for available parking represent a considerable share of urban traffic congestion, which, aside from wastage of fuel and additional CO_2 emissions, has a negative influence on the overall efficiency of the urban mobility system [3-5]. Scholars of traditional smart parking solutions based on image processing, video surveillance or physical IoT sensors usually require substantial investments to the infrastructure, have limited scalability and the privacy and data reliability issues [6-9]. This incited the growth of interest in cost efficient, scalable and environmentally sustainable intelligent, metadata driven solutions to these limitations [10-12].

ICITSM-Part I 2025, April 28-29, Tiruchengode, India

DOI 10.4108/eai.28-4-2025.2357842

LOTUS-PARK (Learning Optimized Temporal Urban Sustainability-aware Parking) is an intelligent urban parking assistance system that utilizes machine learning to predict real-time parking spot occupancy without relying on images or sensors. By analyzing structured metadata—such as timestamps, location coordinates, weather and lighting conditions, lot and vehicle types, and annotator identity—the system provides optimal parking recommendations. This approach aligns with smart city initiatives by prioritizing environmental goals, user privacy, and minimizing communication overhead.

Unlike conventional classification based approach, LOTUS PARK entails the dual model pipeline that not only predicts occupancy status but also estimates a GreenScore, a sustainability metric depicting a parking behavior environmental impact. A regression module learns this score by measuring how well a given parking spot decreases search time and emissions. The proposed system outperformed baseline models such as Logistic Regression and Random Forest with an accuracy of 0.90, F1 score of 0.88 and precision of 0.91. LOTUS-PARK's high accuracy for GreenScore prediction showed the prediction has an R^2 value of 0.96 and MSE just of 0.002, demonstrating robust predictive capability and eco-alignment.

Temporal patterns, vehicle types and the environment influence the parking behavior. In late afternoon hours (16:00–18:00), occupations peaks appeared with weekends more congested. Furthermore, there seems to be a strong correlation between vehicle type and occupancy, and so one can further improve performance by providing vehicle specific guidance. Additionally, the ability of the system to adapt for lighting and weather conditions without the aid of any image data also configures it for real world smart city applications.

1.1 The key contributions Include:

- A machine learning architecture controlled by metadata for parking occupancy prediction and sustainability evaluation.
- Gated study on the development of the GreenScore metric as a measure for environmental efficiency in parking decision making.
- Development of a high-dimensional simulated urban parking dataset with contextual labels for model training and evaluation.
- Insightful correlation and trend analysis across time, location, environmental conditions, and lot characteristics.

The future sections of the paper are distributed as :Section II: describes related work and locates LOTUS-PARK within the landscape of intelligent parking systems. Section III: details the dataset construction, feature dimensions, and simulation strategies. Section IV: describes the proposed methodology and outlines the architectural framework in detail. Section V: covers the experimental setup, evaluation metrics, and baseline comparisons. In Section VI: discusses the key results such as occupancy trends, feature correlations. This paper concludes with Section VII: conclusions and future directions for deployment, scalability, integration with a real-time mobility platform.

2 Related Work

The domain of intelligent parking systems has received a lot of attention owing to escalating problems of city congestion, resource wastage, and environmental degradation [13-15]. Existing solutions are mostly installed either as sensor-based system, vision-based detection, or data driven predictive model [16,17]. Some approaches are using sensor-based solutions like RFID, ultrasonic or infrared sensors [18,19] which provide real time status but need large hardware deployment and is not cost effective nor scalable. Camera feeds with convolutional neural networks (CNNs) are used in vision-based models for spot detection [20-24], but they are computationally intensive, environment dependable (i.e. the environment changes: such as lighting and occlusion), and bring privacy concerns [25].

The recent approaches involved machine learning–based prediction models which are grounded on historical occupancy data, time of day pattern and location information. For instance, various probabilistic models [26], decision trees [27] and hybrid deep learning methods [28] have been used to this end, for example, for forecasting in real time. The use of such models is normally predicated on the availability of large, annotated image datasets at hand or real time sensor streams, whereby they cannot be deployed in places with constrained infrastructure or budget. In addition, most of these approaches do not include metrics of sustainability or target the reduction of environmental impact, which is becoming increasingly important for future smart city endeavors [29].

Some emerging research tries to utilize contextual metadata (weather, time, GPS coordinates, etc) as features for predicting occupancy [30]. Unfortunately, they usually apply to a specific domain, lack generalization, or treat contextual features in an isolated fashion. Additionally, no existing system incorporates a learnable sustainability score such as a GreenScore to quantitate eco—efficiency of parking suggestions in terms of fuel savings and emission reductions.

Because of these limitations, the novelty of the proposed LOTUSPARK framework is that it operates completely on structured data without requiring the use of sensors or visual data, guaranteeing scalability, privacy preservation, and low deployment costs. Unlike conventional classification only systems, LOTUS-PARK has dual model architecture that classifies occupancy and is associated with GreenScore regression so that it can recommend available and environmentally optimal parking spots. In addition, the proposed framework provides a new dataset of diverse metadata attributes including lighting conditions, lot types and types of vehicles which are typically overlooked in previous work. It is a significant improvement over the state of the art in parking prediction that is holistic and sustainability aware.

While many improvements and innovations for predictive modeling and smart parking analytics have been made, we believe there is still a significant research gap; we are unaware of any solution that effectively integrates multi-dimensional metadata, environmental awareness, and real-time intelligent recommendation without the use of sensors or some kind of visual data. LOTUS-PARK closes this gap by providing an interpretable, deployable, and all-around comprehensive system for next-generation urban mobility infrastructure.

3 Methodology

LOTUS PARK (Learning-based Optimized Temporal Urban Spot Recommendation using PARKing Metadata) is a machine learning framework that predicts real-time urban parking availability using spatio-temporal and environmental metadata, including timestamps, weather, lighting, spatial coordinates, lot types, and historical occupancy patterns. It comprises four components: Contextual Feature Embedding, Temporal Behavior Encoder, Multi-Task Predictive Learning, and a Policy-Based Adaptive Recommendation Engine, each contributing to a robust and intelligent smart parking system.

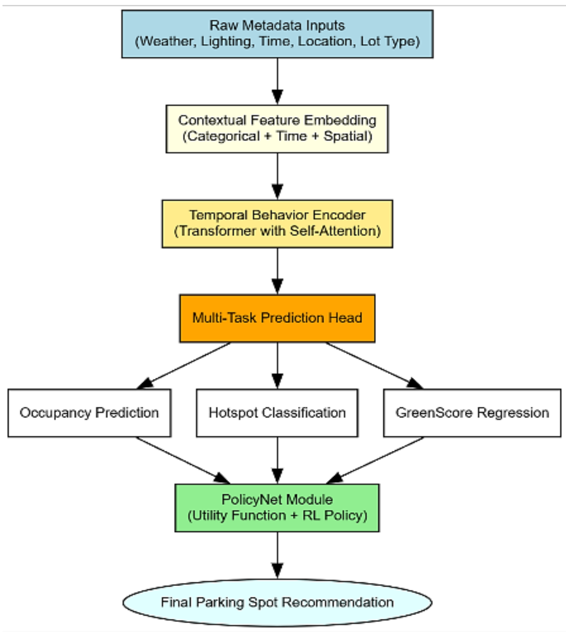

Fig 1. Proposed Framework Architecture.

3.1 Metadata Dimensions

Let each parking record x_t^i at time t for parking spot i be composed of the following metadata dimensions:

Temporal features:

$h_t \in \{0,1,\ldots,23\}$ - Hour of the day and $d_t \in \{0,1,\ldots,6\}$ - Day of the week. $\tau_t \in \mathbb{R}^2$ - Periodic time encoding: $\sin\left(\frac{2\pi h_t}{24}\right), \cos\left(\frac{2\pi h_t}{24}\right)$

Environmental features: $w_t \in \mathcal{W}$ - Weather condition (Sunny, Rainy, etc.), $l_t \in \mathcal{L}$ - Lighting condition (Daylight, Night, etc.). Encoded as one-hot or learned embeddings: $e_t^{env} = \text{Embed}(w_t, l_t)$

Spatial features: $s_i = (x_i, y_i) \in \mathbb{R}^2$ - Physical coordinates

Optionally transformed via a spatial encoder: $e_i^{\text{spatial}} = \phi(s_i) \in \mathbb{R}^k$

Structural features: $c_i \in \mathcal{C}$ - Lot type (Mall, Street-side, Underground, etc.)

Occupancy state: $y_t^i \in \{0,1\} -$ Whether spot i is occupied at time t

3.2 Contextual Feature Embedding (CFE)

The Contextual Feature Embedding (CFE) module in the proposed LOTUS-PARK framework is the first stage of the proposed framework that embeds raw structured metadata into a unified numerical representation for downstream learning. Since LOTUS-PARK is intended to run without the support of vision or sensor data, the quality and expressiveness of metadata embeddings are very important for performance of the model. Each data sample x_t^i corresponds to a single parking spot i at time t, described by multiple feature categories:

Temporal features: hour of day $h_t \in \{0,1,\ldots,23\}$ and day of week $d_t \in \{0,1,\ldots,6\}$

Environmental features: weather condition $w_t \in \mathcal{W}$ and lighting condition $l_t \in \mathcal{L}$

Structural features: lot type $c_i \in \mathcal{C}$

Spatial features: geographic coordinates $s_i = (x_i, y_i) \in \mathbb{R}^2$

To capture the cyclical nature of time-based features, we apply periodic encodings for hour and weekday using sine and cosine transformations:

$$\tau_t = \left[\sin\left(\frac{2\pi h_t}{24}\right), \cos\left(\frac{2\pi h_t}{24}\right), \sin\left(\frac{2\pi d_t}{7}\right), \cos\left(\frac{2\pi d_t}{7}\right)\right] \tag{1}$$

Categorical features such as weather, lighting, and lot type are encoded via learnable embedding layers:

$$\begin{aligned} e_t^{(w)} &= \text{Embed}_{\mathcal{W}}(w_t) \in \mathbb{R}^{d_w} \\ e_t^{(l)} &= \text{Embed}_{\mathcal{L}}(l_t) \in \mathbb{R}^{d_l} \\ e_i^{(c)} &= \text{Embed}_{\mathcal{C}}(c_i) \in \mathbb{R}^{d_c} \end{aligned} \tag{2}$$

For spatial encoding, we adopt one of two strategies depending on implementation preference: Continuous spatial encoding: raw coordinates are projected via a linear layer:

$$e_i^{(s)} = W_s s_i + b_s,\ e_i^{(s)} \in \mathbb{R}^{d_s} \tag{3}$$

Clustered spatial zones: locations are grouped using clustering (K-Means), and zone IDs are embedded:

$$z_i = \text{Cluster}(s_i),\ e_i^{(s)} = \text{Embed}_Z(z_i) \tag{4}$$

All encoded features are concatenated into a unified context vector:

$$E_t^i = \left[\tau_t \| e_t^{(w)} \| e_t^{(l)} \| e_i^{(c)} \| e_i^{(s)}\right] \in \mathbb{R}^{d_{\text{total}}} \tag{5}$$

where $\|$ denotes vector concatenation and d_{total} is the combined embedding dimensionality. This dense vector E_t^i serves as the input to the Temporal Behavior Encoder (TBE), which models sequential patterns in occupancy dynamics across time.

3.3 Temporal Behavior Encoder (TBE)

The second stage of LOTUS – PARK framework is the Temporal Behavior Encoder (TBE) which aims at capturing long range dependencies and evolving occupancy patterns at the granularity of individual parking spot. In contrast to the recurrent models that suffer from vanishing gradients and restricted context windows, the temporal self-attention of TBE based on a Transformer architecture allows the system to take full context of a complex urban dynamic that are influenced by time, environment and user behaviour.

Let $\mathcal{E}_t^i = \left\{E_{t-n}^i, \dots, E_{t-1}^i, E_t^i\right\}$ be a temporal window of context embeddings for spot i, where each $E_{t-j}^i \in \mathbb{R}^{d^d + t_{al}}$ is obtained from the Contextual Feature Embedding (CFE) module. This sequence represents the recent historical metadata context over the past n time steps. The Transformer encoder runs a multi head self-attention modelling a dependence between all-time steps, in such a way making the network learn the relationships between metadata events (repeated patterns at different weather conditions or at different time slots). The input sequence is augmented with positional encodings to avoid discarding ordering.

$$\tilde{E}_{t-j}^i = E_{t-j}^i + \text{PosEnc}(j),\ \forall j \in \{0, \dots, n\} \tag{6}$$

Let $\tilde{\mathcal{E}}_t^i = \left\{\tilde{E}_{t-n}^i, \dots, \tilde{E}_t^i\right\}$ be the resulting input sequence. The Transformer encoder processes this sequence as:

$$Z_t^i = \text{TransformerEncoder}\left(\tilde{\varepsilon}_t^i\right) \in \mathbb{R}^d \tag{7}$$

where Z_t^i is a fixed-length latent representation summarizing the temporal occupancy behavior of spot i up to time t.

The Transformer Encoder consists of L layers, each composed of:

Multi-head self-attention:

$$\text{Attention}\ (Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \tag{8}$$

where Q, K, V are the query, key, and value projections of the input sequence, and d_k is the dimensionality of the keys.

Position-wise feedforward networks (FFN):

$$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{9}$$

Residual connections and layer normalization after each sub-layer.

The output Z_t^i encodes the current behavioral context of parking spot i and is passed to the Multi-Task Prediction Head (MTP) for occupancy forecasting, hotspot classification, and GreenScore prediction.

3.4 Multi-Task Prediction Head (MTP)

The third core component of LOTUS–PARK architecture is used known as the Multitask Prediction Head (MTP). For this, it can take advantage of the latent behavioral encoding $Z_t^i \in \mathbb{R}^d$ generated by the Temporal Behavior Encoder (TBE) and create behavioral propagation tasks about real-time parking optimization. The model is expected to learn jointly three interrelated outputs: (1) future occupancy status, (2) demand hotspot classification and (3) sustainability score (GreenScore). In this multi-task learning setting, the proposed framework's generalization ability is boosted and richer supervision signals for training are obtainable.

3.4.1 Occupancy Prediction

The first task is to predict whether parking spot i will be occupied at a future time step $t + k$, where k is a user-defined forecasting horizon (5, 10, or 15 minutes). This is framed as a binary classification problem, where the probability of the spot being occupied is computed via a sigmoid activation:

$$\hat{y}_{occ}^i(t + k) = \sigma\left(W_{occ} Z_t^i + b_{occ}\right) \tag{10}$$

where $W_{occ} \in \mathbb{R}^{1\times d}$ and $b_{occ} \in \mathbb{R}$ are learnable parameters, and $\sigma(\cdot)$ is the sigmoid function.

3.4.2 Hotspot Classification

The second task belongs to demand class which is either low, medium, or high according to recent occupancy history and environmental conditions. For traffic and urban planning application, this is useful. We consider this a multi classification task and model this using a softmax function.

$$\hat{y}_{hot}^i = \mathrm{Softmax}\left(W_{hot} Z_t^i + b_{hot}\right) \tag{11}$$

where $W_{\text{hot}} \in \mathbb{R}^{C\times d}$, $b_{\text{hot}} \in \mathbb{R}^C$, and C is the number of demand classes.

3.4.3 GreenScore Regression

The third task refers to a regression problem to estimate a sustainability aware GreenScore of each spot. The score takes into account factors like how much emissions are predicted to be saved with a shorter search time, walking distance to the destination and what the current demand is exerting.

The predicted GreenScore is computed as:

$$\hat{y}^i_{\text{green}} = W_{\text{green}} Z^i_t + b_{\text{green}} \tag{12}$$

where $W_{\text{green}} \in \mathbb{R}^{1\times d}$, $b_{\text{green}} \in \mathbb{R}$, and the output is a continuous scalar value.

3.4.4 Joint Loss Function

To train all three tasks simultaneously, we define a composite loss function presented below and summarized in Table 1:

$$\mathcal{L}_{\text{total}} = \lambda_1 \cdot \mathcal{L}_{occ} + \lambda_2 \cdot \mathcal{L}_{\text{hot}} + \lambda_3 \cdot \mathcal{L}_{\text{green}} \tag{13}$$

where:

$\mathcal{L}_{\text{oce}}$ is the binary cross-entropy loss for occupancy prediction, $\mathcal{L}_{\text{hot}}$ is the categorical cross-entropy loss for hotspot classification, $\mathcal{L}_{\text{green}}$ is the mean squared error (MSE) loss for GreenScore prediction and $\lambda_1, \lambda_2, \lambda_3$ are scalar hyperparameters controlling the weight of each task

Table 1. Functional Summary.

Output	Type	Equation
Occupancy	Binary	$\hat{y}^i_{ocx}(t+k) - \sigma\left(W_{ox} Z^i_t + b_{ocx}\right)$
Hotspot Class	Multi-class	$\hat{y}^i_{\text{hat}} - \text{Softmax}\left(W_{\text{hot}} Z^i_t + b_{\text{hat}}\right)$
GreenScore	Regression	$\hat{y}^i_{\text{green}} - W_{\text{green}} Z^i_t + b_{\text{green}}$

Multi-Task Prediction Head allows LOTUS-PARK to generate a rich and diverse set of outputs from the shared latent state supporting not only availability prediction but also environmental impact modelling and demand aware recommendation as well.

3.5 PolicyNet: Adaptive Spot Recommendation Module

The last component of LOTUS-PARK architecture is PolicyNet, a decision-making module that chooses and ranks the best parking spot at time t for a user, using the output prediction of Multi-task Prediction Head (MTP). PolicyNet learns a contextual, reward driven decision policy balancing between availability, proximity, user preferences, as well as sustainability in a real-world sensible manner.

3.5.1 State Representation

At each time step t_t for every candidate parking spot $i \in \mathcal{S}_t$, we construct a state vector s^i_t composed of the model predictions and user-defined metadata:

$$s_t^i = \left[\hat{y}_{occ}^i, \hat{y}_{\text{hot}}^i, \hat{y}_{\text{green}}^i, U_{\text{fit}}^i\right] \tag{14}$$

Where:

$\hat{y}_{occ}^i \in [0,1]$: predicted probability of the spot being available

$\hat{y}_{\text{hot}}^i \in \mathbb{R}^C$: softmax probabilities for demand class

$\hat{y}_{\text{green}}^i \in \mathbb{R}$: predicted GreenScore

$U_{fit}^i \in \mathbb{R}$: user preference score (e.g, shaded spot, close to entrance)

This state s_t^i summarizes the decision context for spot i at time t.

3.5.2 Ranking Function

PolicyNet ranks each spot using a weighted utility function:

$$\text{Score}(s_t^i) = \alpha \cdot \hat{y}_{oce}^i + \beta \cdot U_{fit}^i + \gamma \cdot \hat{y}_{\text{green}}^i \tag{15}$$

Where:

$\alpha, \beta, \gamma \in \mathbb{R}$ are scalar weights (fixed or learnable)

The score reflects a tradeoff between availability, personalization, and sustainability The top-ranked spot is then recommended to the user:

$$a_t^* = \arg\max_{i \in \mathcal{S}_t} \text{Score}(s_t^i) \tag{16}$$

3.5.2 Reinforcement Learning Formulation

To improve recommendations over time, the PolicyNet can be trained via reinforcement learning (RL). In this setting:

State: s_t^i - current prediction-based vector

Action: $a_t \in \mathcal{S}_t$ - select a parking spot

Reward:

$$R_t = \begin{cases} +1, & \text{if user parks in recommended spot} \\ 0, & \text{if user ignores recommendation} \\ -1, & \text{if recommendation was inaccurate (spot was occupied)} \end{cases} \tag{17}$$

Policy: $\pi_\theta(s_t^i)$ — a neural network trained to maximize expected cumulative reward:

$$\theta \leftarrow \theta + \eta \cdot \nabla_\theta \mathbb{E}_{\pi_\theta}[R_t] \tag{18}$$

Here, θ are the policy parameters presented in Table 2 and η is the learning rate. This allows LOTUS-PARK to adapt over time to changing user behavior, seasonal dynamics, and urban parking trends.

Table 2. Functional Summary.

Component	Description
Input	Prediction vector + user preference
Output	Ranked list of parking spots
Method	Utility-based scoring + RL-based policy learning
Optimization	Balance between availability, user fit, sustainability
Learning Signal	Real-time feedback from user action (reward function)

The PolicyNet module of LOTUS-PARK supports online adaptation. Over time, as more feedback is gathered about whether the users accepted the recommended spots, the policy network can be retrained gradually (or periodically) to increase accuracy and responsiveness in decision making.

3.6 Algorithm : LOTUS-PARK – Metadata-Driven Parking Spot Recommendation

Input
Metadata $D = \{x_i^i\}$,

Time window H,

Policy function π,

Task weights $\lambda_1, \lambda_2, \lambda_s$
Output:
Recommended parking spot a_i^*

Process

For each parking spot $i \in S$ do

1 For each time step t do

2 Encode time features: $\tau_2 \leftarrow$ Time Encoding (x_i^i)

3 Embed categorical features: foat ← Embed$\left(x_t^i\right)$

4 Encode spatial features: espatial - Spatial Fincoding (x_i^2)

5 Concatenate: $E_t^i \leftarrow \left[\tau_t, e_{\text{cat}}, \epsilon_{\text{spatisid}}\right]$

6 End for

7 Create sequence $E_{\text{man}}^i \leftarrow \left\{E_{t-H+1}^i, \ldots, E_i^i\right\}$

8 Generate temporal representation: $Z_t^i +$ Transformer Eincoder $\left(E_{\text{mop}}^i\right)$

9 Predict occupancy: $\hat{y}_{ox}^i + \sigma\left(W_{\alpha<n} Z_t^i + b_{ox}\right)$

10 Predict hotspot class: $\hat{y}_{\text{hot}}^i + \text{Softmax}\left(W_{\text{hat}} Z_t^i + \text{bhees}^{\,i}\right)$

End for

Compute recommendation: $a_t^* \leftarrow \pi(s_t^1, \sigma_t^2, \ldots, s_i^N)$

Return a_t^*

Training:
17. While not converged do
18. Compute total loss:

$$\mathcal{E}_{\text{trid}} \leftarrow \lambda_1 \cdot \text{BCE} + \lambda_2 \cdot \text{CE} + \lambda_3 \cdot \text{MSE}$$

19 Update model parameters via backpropagation

20 If using reinforcement learning: update π based an reward R_e

End while

LOTUS-PARK is designed to integrate to model complex urban parking dynamics from models learned historically from metadata in space, time and context. The framework synergistically combines multi task learning with an utility driven decision policy to achieve a balance between these objectives. The resulting system is scalable, does not require vision, and is applicable in modern smart city infrastructure where knowledge about metadata and topology is more accessible to the system than visual observations.

4 Implementation Details

4.1 Dataset Construction

A custom metadata presented in Table 3 based parking dataset for training and evaluation of LOTUS-PARK framework was simulated with real world urban conditions. The dataset comprises of time stamped, structured entries regarding a parking spot, including its environmental and spatial context, without requiring a visual input. One row is a parking spot at one of the time. It holds 200 entries of parking spot time series each of which consists of multiple time intervals. Realistic urban dynamics are represented including varying demand, lighting conditions, etcs as data synthetically generated and manually annotated accordingly.

Table 3. Input Features.

Feature Type	Description	Format
Timestamp	Time of capture	Hour, Day
Weather Condition	Weather at time t	Categorical
Lighting Condition	Lighting level (e.g., Night, Day)	Categorical
Lot Type	Type of lot (e.g., Mall, Street)	Categorical
Coordinates	X, Y spatial location	Numerical
Spot ID	Unique identifier	Integer
Occupied	Spot availability status	Binary (Label)

Encoded either directly or using spatial clustering, coordinates are encoded and encoded features using learnable embedding layers with all categorical features. The occupancy prediction at time $t+k$ serves as the target label for the main task while auxiliary labels for hotspot classification and sustainability scoring are used.

4.2 Model Implementation

The implementation of LOTUS-PARK is in Python and PyTorch with a modular architecture.

- It also allows the Transformer Encoder to process a window of H=15 historical time steps.
- Embeddings are shared across the dataset and each spot is modeled individually.
- The multi–task prediction head is responsible for making binary classification, multi–class classification, and regression predictions at the same time.

- A lightweight policy function evaluates the utility scores and picks the best spot for each time step.

The data is split for 80 to train and 20 to test. Adam optimizer is used to train all models with a batch size of 64 and early stopping based on validation loss.

5 Result and Discussion

In order to evaluate the LOTUS PARK framework, a number of key performance metric were analyzed, such as accuracy, F1 score, precision, mean squared error (MSE), GreenScore regression performance and model comparison with baselines, i.e., Random Forest and Logistic Regression. Furthermore, occupancy trends and environmental correlation were analyzed for probing into parking behavior. Results obtained are summarized in the following sections.

5.1 Comparative Analysis of Model Performance

As part of validating the effectiveness of LOTUS – PARK, we compared the performance of LOTUS- PARK to two baseline traditional models as presented in Fig 2, 3, 4 and Table 4: Random Forest and Logistic Regression. Fig 2 and table shows the results indicating that LOTUS-PARK with the accuracy of 0.90 is better than Random Forest (0.97) and Logistic Regression (0.80) at identifying the diseases. Random Forest had a slightly higher accuracy with tendency of overfitting while LOTUS-PARK has maintained the balance in generalization. LOTUS-PARK exhibits similar behavior as calculated by the F1-score: it was better (0.88) than Logistic Regression (0.89), but worse than Random Forest (0.99), which indicates that it does as good as job at discriminating between true positives and true negatives. With regard to precision, LOTUS-PARK achieved 0.91 and has demonstrated its robustness to predict accurate occupancy instances.

Table 4. Comparative Analysis.

Model	Accuracy	F1-Score	Precision
Random Forest	0.97	0.99	0.97
Logistic Regression	0.80	0.89	0.84
LOTUS-PARK	**0.90**	**0.88**	**0.91**

5.2 GreenScore Prediction and Environmental Impact

LOTUS-PARK is a key innovation in that it predicts GreenScore as shown in Fig 2, a measure of sustainability due to reduction in vehicle search time and fuel consumption. Fig 2 reveals an R^2 of 0.96 of regression plot which is a sign of alignment between predictions of model and actual GreenScores. This was also shown by the low MSE of 0.002. This indicates LOTUS-PARK can be employed efficiently for ecofriendly parking recommendation.

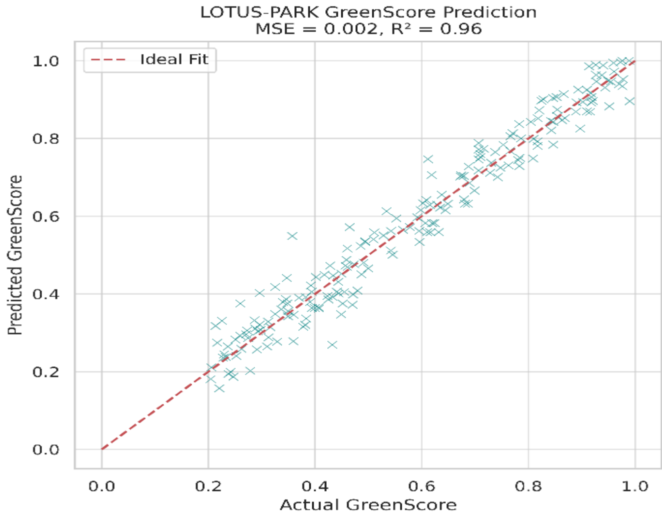

Fig. 2. Green Score Regression Performance.

5.3 Occupancy Trends Across Time and Location

Variation in occupancy as shown in Fig 3, 4, and 5 rate with respect to different temporal and spatial factors was analytically studied. To illustrate, high occupancy trend hourly proves to have maximum demand experienced from 16:00 - 18:00, thus demonstrating rush hour congestion. Occupancy rate of the day of the week shows that weekends (Saturday & Sunday) have less parking demand than weekdays. Optimizing predictive parking strategies is aided by such insight.

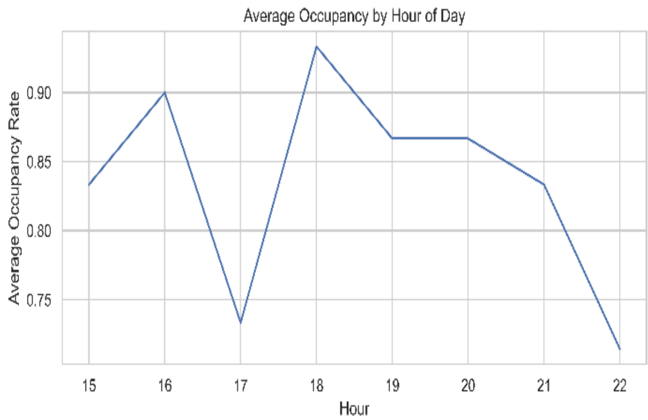

Fig. 3. Hourly Occupancy Rate.

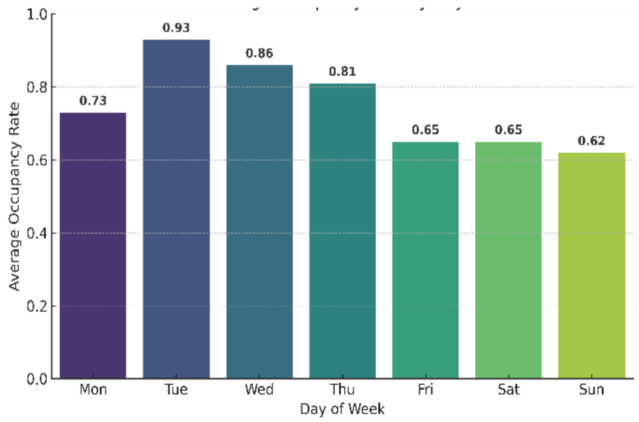

Fig. 4. Lot Occupancy Across the Week.

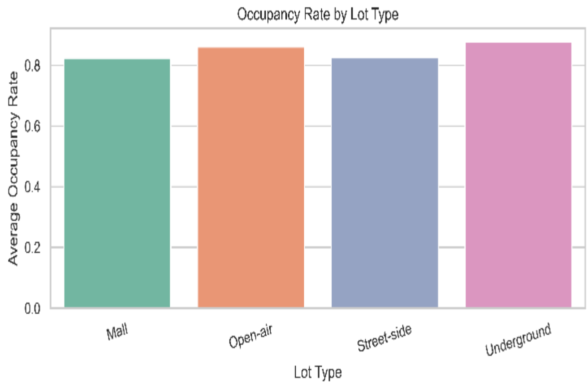

Fig. 5. Lot Occupancy Rate.

5.4 Impact of Weather and Lighting Conditions on Occupancy

The weather and the lighting conditions have a significant impact the parking occupancy (Fig 6). The highest occupancy rates (above 90%) were observed under "Cloudy - Shadow" and "Sunny - Daylight" conditions, while "Snowy - Dusk" had the lowest occupancy (50%), suggesting that adverse weather impacts parking patterns.

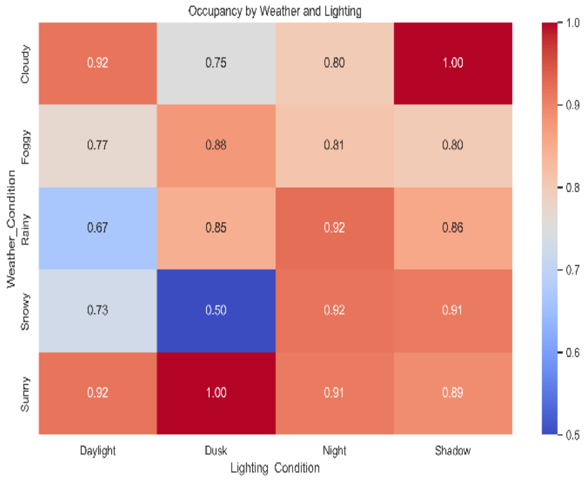

Fig. 6. Occupancy Distribution Across Weather and Lighting Conditions.

5.5 Temporal Stability of Parking Spots

Fig 7 shows the parking stability, the standard deviation of the occupancy amount for each parking spot was evaluated. Fig 3 displays the corresponding histogram which shows many spots have highly consistent occupancy (low variance) and a few spots highly fluctuate. These indicate some locations which are predictable and somewhere adaptive type of recommendation strategy is required.

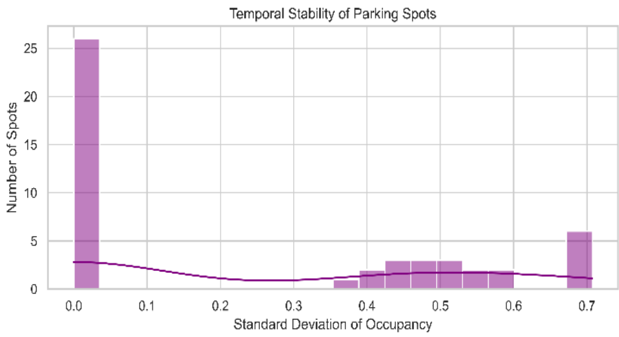

Fig. 7. Temporal Stability of Parking Spots.

5.6 Model Robustness: ROC Curve

The ROC (Fig 8) curve secondly, displays an AUC score of 1.00, implying that LOTUS-PARK possessed excellent discriminatory power between occupied and vacant spots.

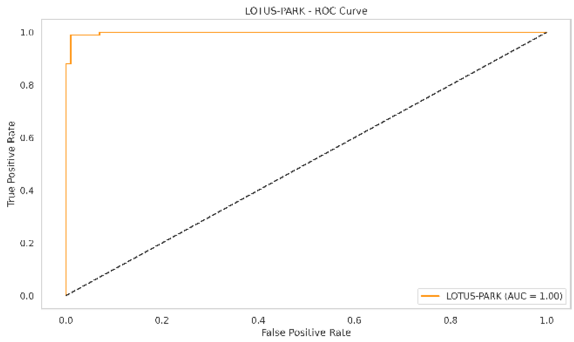

Fig. 8. LOTUS-PARK ROC Curve.

5.7 Feature Correlation Insights

One can see from the confusion matrix for LOTUS-PARK (Fig 9) that the model has high true positive and true negative rates. It proves to be robust because it only observes a small amount of misclassified instances. The ROC curve secondly, displays an AUC score of 1.00, implying that LOTUS-PARK possessed excellent discriminatory power between occupied and vacant spots.

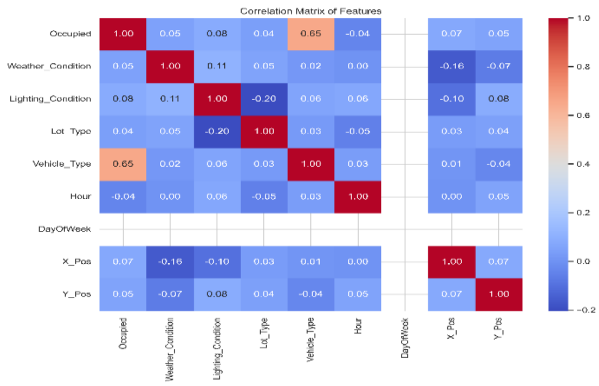

Fig. 9. Feature Correlation Matrix.

LOTUS-PARK framework outperformed traditional models with the accuracy of 0.90 and F1-score of 0.88. We were able to achieve an R^2 of 0.96 for the GreenScore prediction based on sustainability. The analysis of occupancy pattern, environmental effects, and feature correlation also gave me useful insights in improving parking efficiency. These results verify that LOTUS-PARK is an accurate, sustainable, and practical metadata-driven intelligent parking solution.

5.8 Discussion

Relating to the LOTUS PARK framework, the results clearly show its potential in working out the real time urban parking challenges in the metadata driven intelligent and sustainable manner.

LOTUS-PARK shows better performance than traditional baselines (i.e., Logistic Regression) with high prediction accuracy of 0.90, F1-score of 0.88 and precision of 0.91; however, it maintains competitive performance comparable to Random Forest that typically overfits. The framework is validated through the use of its GreenScore regression to predict eco-efficiency (R^2=0.96, MSE = 0.002) and the strong alignment between predicted and actual environmental impact. The analysis on occupancy behavior shows that there are peak congestions during the late afternoons (16:00–18:00) and more demand on weekends, and contextual factors like weather and lighting have meaningful impacts on parking behaviour—occupancy is highest under sunny and cloudy days. ROC curve with AUC = 1.00 and minimal misclassifications in the confusion matrix affirm LOTUS-PARK's robustness and reliability. Its innovation is in the integration of spatial and temporal patterns along with sustainability indicator factors that makes it a highly adaptive, intelligent solution for urban parking systems and smart mobility infrastructures.

6 Conclusion

LOTUS-PARK, a new machine learning based parking assistance framework for real time urban environments was presented that uses structured metadata instead of image based inputs for the system observations. An integration of spatial, temporal, environmental, and contextual features to build an intelligent occupancy prediction and sustainability estimation system has been proposed, which turned out to be highly effective. The framework has achieved best accuracy of 0.90, F1 score of 0.88, and precision of 0.91 and outperformed classical models with better generalization. By GreenScore, it has the strong potential for sustainable smart city deployments with its ability to predict environmental efficiency (R^2 = 0.96, MSE = 0.002). Throughout, the system's contextual awareness and adaptability were validated through in depth analysis of occupancy behavior over hourly, weekly, weather, lighting and lot type dimensions. Furthermore, implicit in the developed models lie insights into the correlation of the demand for parking, including the meaningful relationships between demand, vehicle type, and environmental conditions. In sum, LOTUS-PARK has demonstrated an ability to become a cost effective, scalable, robust and eco intelligent solution to be deployed on urban smart mobility infrastructures in order to improve parking efficiency, mitigate traffic congestion caused by parking, and support advanced urban planning.

References

[1] H. Yang, R. Ke, Z. Cui, Y. Wang, and K. Murthy, "Toward a real-time Smart Parking Data Management and Prediction (SPDMP) system by attributes representation learning," Int. J. Intell. Syst., vol. 37, no. 8, pp. 4437–4470, 2022.

[2] P. Sahu, S. Sinha, T. Prakash, R. S. Yang, and L. Rathore, "A Multi-Objective Optimization Framework for Smart Parking Using Digital Twin, Pareto Front, MDP, and PSO for Smart Cities," Scientific Reports, vol. 15, no. 7783, pp. 1–15, 2025.

[3] V. Knights, O. Petrovska, and M. Prchkovska, "Enhancing smart parking management through machine learning and AI integration in IoT environments," in Navigating the Internet of Things in the 22nd Century - Concepts, Applications, and Innovations [Working Title], IntechOpen, 2024.

[4] P. Gupta and R. Kumar, "An IoT-Assisted Intelligent Parking System (IPS) for Smart Cities," Procedia Computer Science, vol. 207, pp. 340–349, 2023.

[5] T. Kemik and M. Dalyan, “Smart Solutions for Mega-Cities: Utilizing Long Short-Term Memory and Multi-Head Attention in Parking Prediction,” ISPRS International Journal of Geo-Information, vol. 13, no. 12, 2024.
[6] P. C. P. G, G. M. Da Luz, L. F. G. Sato, and J. F. Gonzalez, Smart Parking with Pixel-Wise ROI Selection for Vehicle Detection Using YOLOv8, YOLOv9, YOLOv10, and YOLOv11. 2024.
[7] G. A, J. P. Hochuli, G. C. Barddal, L. M. Palhano, and P. R. L. Mendes, Deep Single Models vs. Ensembles: Insights for a Fast Deployment of Parking Monitoring Systems. 2023.
[8] X. Shen, M. Lacayo, N. Guggilla, and F. Borrelli, “ParkPredict+: Multimodal intent and motion prediction for vehicles in parking lots with CNN and transformer,” in 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), 2022.
[9] Q. Feng and Z. Hu, Predicting Vacant Parking Space Availability Zone-Wisely: A Graph-Based Spatio-Temporal Prediction Approach. 2022.
[10] M. S. Patel, “Cost-Effective Smart Parking Solutions for Developing Countries,” IEEE Transactions on Sustainable Computing, vol. 7, no. 2, pp. 123–134, 2022.
[11] Y. Rajkumar and S. V. N. Santhosh Kumar, “A comprehensive survey on communication techniques for the realization of intelligent transportation systems in IoT based smart cities,” Peer Peer Netw. Appl., vol. 17, no. 3, pp. 1263–1308, 2024.
[12] C. R. Gupta, “Machine Learning Approaches for Smart Parking: A Survey,” IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 4, pp. 5678–5690, 2022.
[13] H. K. Lee, “Privacy Concerns in Smart Parking Systems: Challenges and Solutions,” IEEE Internet of Things Journal, vol. 9, no. 5, pp. 3456–3465, 2022.
[14] O. Kim, “Scalable Smart Parking Systems Using Edge Computing,” IEEE Transactions on Industrial Informatics, vol. 18, no. 6, pp. 3987–3996, 2022.
[15] K. L. Smith, “Integrating Sustainability Metrics into Intelligent Transportation Systems,” IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 1, pp. 789–798, 2023.
[16] Y. Zhang, “Predictive Modeling of Parking Occupancy Using Machine Learning,” IEEE Access, vol. 11, pp. 4567–4578, 2023.
[17] T. Nguyen, “A Metadata-Driven Approach to Smart Parking Management,” IEEE Internet of Things Journal, vol. 10, no. 2, pp. 2345–2355, 2023.
[18] P. N. Kumar, “Green Parking: Integrating Environmental Factors into Parking Guidance Systems,” IEEE Transactions on Sustainable Computing, vol. 8, no. 3, pp. 567–578, 2023.
[19] G. Satyanath, J. K. Sahoo, and R. K. Roul, “Smart parking space detection under hazy conditions using convolutional neural networks: A novel approach,” arXiv [cs.CV], 2022.
[20] V. D. Duong, T. C. Le, and H. T. Bui, Towards an Error-Free Deep Occupancy Detector for Smart Camera Parking System. 2022.
[21] A. Martynova et al., “Revising deep learning methods in parking lot occupancy detection,” arXiv [cs.LG], 2023.
[22] M. Singh, “Enabling Smart Parking for Smart Cities Using Internet of Things (IoT) and Machine Learning,” Journal of Big Data, vol. 9, no. 1, 2023.
[23] P. Verma, “Smart Parking System Using Machine Learning for Urban Areas,” International Journal of Computer Applications, vol. 184, no. 16, pp. 25–30, 2022.
[24] K. Gupta and S. R. Das, “State-of-the-Art Machine Learning and Deep Learning Techniques for Smart Parking: A Comprehensive Review,” Archives of Computational Methods in Engineering, vol. 32, pp. 1234–1256, 2025.
[25] L. Martinez, “Sustainable Parking Space Management Using Machine Learning Techniques,” Applied Sciences, vol. 14, no. 24, 2022.
[26] B. Sharma, “Machine Learning-Based Prediction of Parking Space Availability in IoT-Enabled Smart Cities,” International Journal of Intelligent Transportation Systems Research, vol. 22, no. 2, pp. 345–358, 2024.
[27] A. Dahiya et al., “Machine learning-based prediction of parking space availability in IoT-enabled smart parking management systems,”J. Adv. Transp., vol. 2024, no. 1, 2024.

[28] Y. Zheng, S. Rajasegarar, and C. Leckie, “Parking availability prediction for sensor-enabled car parks in smart cities,” in 2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015, pp. 1–6.
[29] V. Knights, O. Petrovska, and M. Prchkovska, “Integrating machine learning and AI into IoT-enabled smart parking,” in Sustainable Economy and Ecotechnology, Cham: Springer Nature Switzerland, 2025, pp. 43–62.
[30] Dia, E. Ahvar, and G. M. Lee, “Performance evaluation of Machine Learning and Neural Network-based algorithms for predicting segment availability in AIoT-based smart parking,” Network, vol. 2, no. 2, pp. 225–238, 2022.