



ELECTRIC VEHICLE CHARGING STATION LOCATER AND SLOT BOOKING SYSTEM USING ARTIFICIAL INTELLIGENCE

Dr. S. Nagarajan,*

Swarnandhra College of Engineering and technology, Narsapur, Andhra Pradesh,
Associate Professor, Department of Computer Science Engineering.

Mrs. S.P. Sudha,**

Swarnandhra College of Engineering and technology, Narsapur, Andhra Pradesh,
Assistant Professor, Department of Artificial Intelligence and Machine Learning

ABSTRACT:

The Electrification of the transportation sector is one of the viable solutions in minimizing the carbon emission over the atmosphere. Due to the rapid growth of electric vehicles in the market, electric vehicle charging stations are in demand among electric vehicle users. As charging of electric vehicles takes a long time, prebooking of charging slots can be done to avoid the waiting time. Artificial intelligence leverages computers and machines to mimic the artificially problem-solving and decision-making capabilities of the human mind. In general AI is a field, which combines computer science and datasets, to enable problem-solving. In the existing system, the requests are handled on First Come First Serve basis. The complexity of using FCFS is its long wait times. In the proposed system, the complexity of FCFS is overcome by using the round-robin scheduling algorithm which allocates slots based on predefined time intervals. Therefore the availability of free booking slots is known to the user. KNN Algorithm is used to find the nearest charging station. The proposed system will result in the reduction of waiting time and good accuracy.

Keywords: Electric Vehicle Charging Station (EVCS), Slot booking, GPS, Artificial Intelligence, Application, Map.

INTRODUCTION:

ARTIFICIAL INTELLIGENCE:

Developing an AI-powered EV charging station locator and slot booking system involves complex integration of various technologies and functionalities. Collaboration with charging network providers, maintaining data accuracy, and ensuring a seamless user experience are crucial aspects of such a system. Artificial intelligence is the simulation of human intelligence processes by machines, especially computer systems. Specific applications of AI include expert systems, natural language processing speech recognition and machine vision. As the hype around AI has accelerated, vendors have been scrambling to promote how their products and services use it. Often, what they refer to as AI is simply a component of the technology, such as machine learning.

Artificial Intelligence (AI) plays a significant role in enhancing various aspects of an Electric Vehicle (EV) charging station locator and slot booking system. AI algorithms can analyze historical charging station usage data to predict future demand patterns. This helps in optimizing station availability and planning expansions.

AI-powered EV charging station locator and slot booking system

AI requires a foundation of specialized hardware and software for writing and training machine learning algorithms. Creating an electric vehicle (EV) charging station locator and slot booking system using artificial intelligence involves several key components and functionalities. No single programming language is synonymous with AI, but Python, R, Java, C++ and Julia have features popular with AI developers. In general, AI systems work by ingesting large amounts of labeled training data, analyzing the data for correlations and patterns, and using these patterns to make predictions about future states. In this way, a chatbot that is fed examples of text can learn to generate lifelike exchanges with people, or an image recognition tool can learn to identify and describe objects in images by reviewing millions of examples. New, rapidly improving generative AI techniques can create realistic text, images, music and other media.

Data Collection and Integration

- Charging Station Database: Gather information on available charging stations, their locations, types of chargers, availability, and supported vehicles.
- Real-time Updates: Utilize APIs or direct connections to charging networks for real-time availability and status updates.

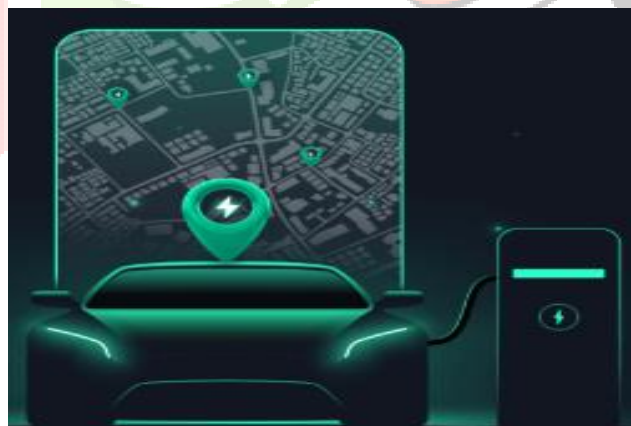


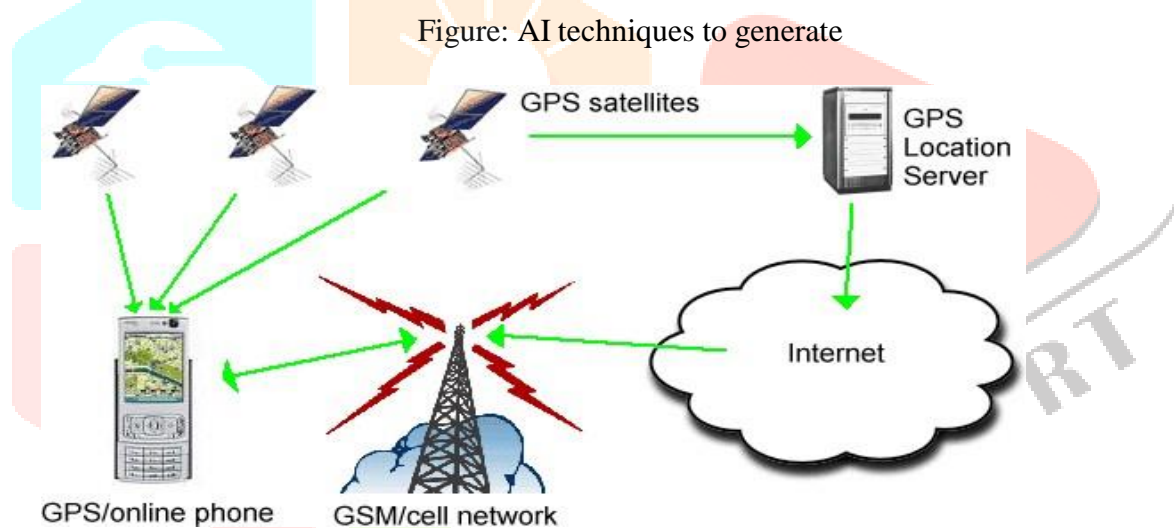
Figure: AI-powered EV charging station locator and slot booking system

AI programming focuses on cognitive skills that include the following:

- **Learning.** This aspect of AI programming focuses on acquiring data and creating rules for how to turn it into actionable information. The rules, which are called algorithms, computing devices with step-by-step instructions for how to complete a specific task.
- **Reasoning.** This aspect of AI programming focuses on choosing the right algorithm to reach a desired outcome.
- **Self-correction.** This aspect of AI programming is designed to continually fine-tune algorithms and ensure they provide the most accurate results possible.
- **Creativity.** This aspect of AI uses neural networks, rules-based systems, statistical methods and other AI techniques to generate new images, new text, new music and new ideas.

NATURAL LANGUAGE PROCESSING:

Natural language processing (NLP) refers to the branch of computer science and more specifically, the branch AI concerned with giving computers the ability to understand text and spoken words in much the same way human beings can. NLP combines computational linguistics rule-based modeling of human language with statistical, machine learning, and deep learning models. Together, these technologies enable computers to process human language in the form of text or voice data and to ‘understand’ its full meaning, complete with the speaker or writer’s intent and sentiment. NLP drives computer programs that translate text from one language to another, respond to spoken commands, and summarize large volumes of text.



Rapidly even in real time. There’s a good chance you’ve interacted with NLP in the form of voice-operated GPS systems, digital assistants, speech-to-text dictation software, customer service chatbots, and other consumer conveniences. Natural Language Processing (NLP) is a branch of artificial intelligence that focuses on enabling computers to understand, interpret, and generate human language in a way that is both meaningful and contextually relevant. In the context of an Electric Vehicle (EV) charging station locator and slot booking system, But NLP also plays a growing role in enterprise solutions that help streamline business operations, increase employee productivity, and simplify mission-critical business processes.

GLOBAL POSITIONING SYSTEM (GPS):

GPS (Global Positioning System) is a satellite-based navigation system. It provides time and location-based information to a GPS receiver, located anywhere on or near the earth surface. GPS works in all weather conditions provided there is an unobstructed line of sight communication with 4 or more GPS satellites. GPS is managed by the US Air Force. A GPS operates independently of the user’s internet connection or telephone signal. However, their presence increases the effectiveness of GPS positioning. GPS was initially developed by the US government for military purpose, but currently, anyone with a GPS receiver can receive radio signals from GPS satellites.

The need for installing EV charging stations in several areas is rising as they become more and more popular. The majority of these stations are part of a charging network, so customers may access them and find EV charging station finder apps on their mobile devices. Customers now can easily find an EV charging station nearby thanks to the introduction of these apps. Although there are many EV Charging Station finding app they themselves have their own drawbacks and to overcome the issues in existing system we have proposed some objectives,

The objectives of this project include,

- An application which shows nearby charging station where, location is given as input.
- Customers can also reserve a slot in advance for charging the Electric Vehicle
- Customers can prefer their own time slot according to their convenience.
- While viewing the location in the map we can also able to see the slot availability.
- Customers can also cancel their reservation at any time.
- Can make payments directly through the apps used by electric vehicle charging stations.
- Admin has the privilege in updating locations and their slot availability

TECHNIQUE USED:

The K-Nearest Neighbors (K-NN) algorithm is a simple yet effective supervised machine learning algorithm used for both classification and regression tasks. It's a non-parametric and instance-based learning method, meaning it doesn't make assumptions about the underlying data distribution and stores the entire training dataset for predictions.

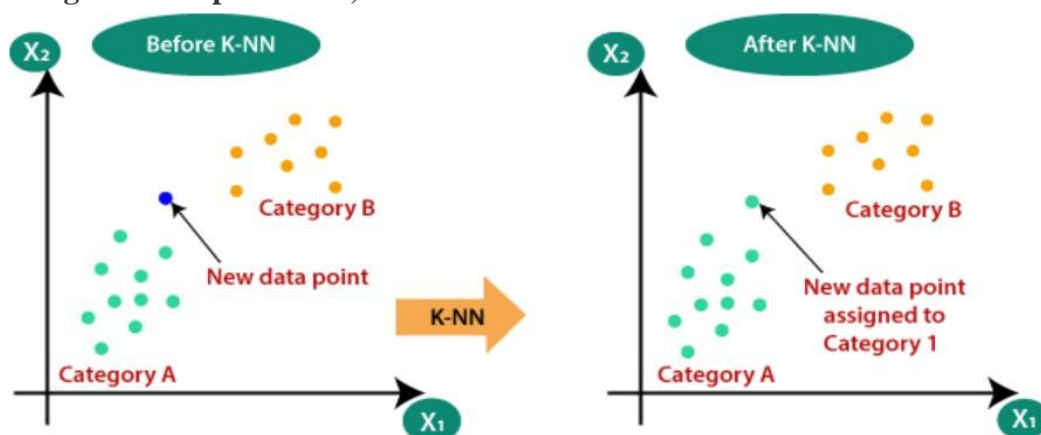
- **Classification:** For a given query instance, the K-NN algorithm identifies the 'K' nearest neighbors in the training dataset based on a chosen distance metric (such as Euclidean distance, Manhattan distance, etc.).
- **Regression:** For regression tasks, K-NN computes the average or weighted average of the 'K' nearest neighbors' target values to predict the value for the query instance.

Algorithm 1: K-Nearest Neighbour:

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique. It assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. It stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.

K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems. K-NN is a non- parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset. K- NN algorithm at the training phase just stores the dataset and when it gets new data, and then it classifies that data into a category that is much similar to the new data.

The K-NN working can be explained as,



- **Step-1:** Select the number K of the neighbours
- **Step-2:** Calculate the Euclidean distance of K number of neighbours
- **Step-3:** Take the K nearest neighbours as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbours, count the number of the datapoints in each category.
- **Step-5:** Assign the new data points to that category for which the number of the Neighbour is maximum.

There is no particular way to determine the best value for "K", so we need to try some values to find the best out of them. The most preferred value for K is 5. A very low value for K such as K=1 or K=2, can be noisy and lead to the effects of outliers in the model. Large values for K are good, but it may find some difficulties.

Algorithm 2: Round robin Scheduling:

Round Robin (RR) scheduling is a CPU scheduling algorithm used in operating systems that handles the way processes are executed in a system. It is a preemptive algorithm that allocates a fixed time slice to each process, allowing them to execute for a specified quantum (time quantum) before switching to the next process in the queue. Round Robin CPU Scheduling is the most important CPU Scheduling Algorithm which is ever used in the history of CPU Scheduling Algorithms. Round Robin CPU Scheduling uses Time Quantum (TQ). The Time Quantum is something which is removed from the Burst Time and lets the chunk of process to be completed. Time Sharing is the main emphasis of the algorithm. Each step of this algorithm is carried out cyclically. The system defines a specific time slice, known as a time quantum. First, the processes which are eligible to enter the ready queue enter the ready queue. After entering the first process in Ready Queue is executed for a Time Quantum chunk of time. After execution is complete, the process is removed from the ready queue.

Even now the process requires some time to complete its execution, then the process is added to Ready Queue. The Ready Queue does not hold processes which already present in the Ready Queue. The Ready Queue is designed in such a manner that it does not hold non unique processes. By holding same processes Redundancy of the processes increases. After, the process execution is complete; the Ready Queue does not take the completed process for holding.

User Interface:

- Mobile/Web App: Develop an intuitive interface for users to access the system.
- Search and Map Integration: Allow users to find nearby charging stations using maps or search functionalities.
- User Profiles: Users can create profiles, save preferences, and manage bookings.

AI-Powered Features:

- Optimized Route Planning: Use AI algorithms to suggest the most efficient route considering available charging stations, traffic conditions, and user preferences.
- Predictive Availability: Utilize machine learning to predict charging station availability based on historical data and current usage patterns.

Slot Booking System:

- Real-time Booking: Allow users to check the availability of charging slots in real-time and book slots in advance.
- Notification System: Send alerts and reminders about booked slots, charging completion, or any changes in station status.

Integration of Payment Gateway:

Secure Transactions: Enable users to make payments for charging services directly through the app/platform.

Integrating a payment gateway into a system involves connecting your application or website to a service that facilitates online transactions securely.

Choose a Payment Gateway Provider:

1. Research and select a reputable payment gateway provider that suits your business needs. Popular options include PayPal, Stripe, Square, Authorize.Net, etc.

2. Set Up Merchant Account:

Create an account with the selected payment gateway provider. This involves registering as a merchant and providing necessary business information.

3. Obtain API Keys or SDKs:

Once you have an account, the payment gateway provider will give you API keys, SDKs (Software Development Kits), or other tools necessary for integration.

4. Development and Integration:

Integrate the payment gateway's API or SDK into your application's codebase. This involves programming to enable communication between your application and the payment gateway provider's servers.

Ensure encryption and security measures are implemented to protect sensitive user payment data.

5. Test Transactions:

Perform test transactions using sandbox or test environments provided by the payment gateway to ensure the integration works correctly without processing actual payments.

6. Compliance and Security:

Adhere to PCI-DSS (Payment Card Industry Data Security Standard) compliance to ensure the security of cardholder data.

Implement additional security measures such as tokenization and SSL encryption to safeguard sensitive information.

7. User Interface:

Create a user-friendly interface for payment processing within your application or website. This might include checkout forms, payment buttons, or a dedicated payment page.

8. Error Handling and Notifications:

Implement error handling mechanisms to deal with failed transactions, network issues, or other errors that may occur during payment processing.

Set up notifications or alerts to inform users about successful transactions and provide confirmation details.

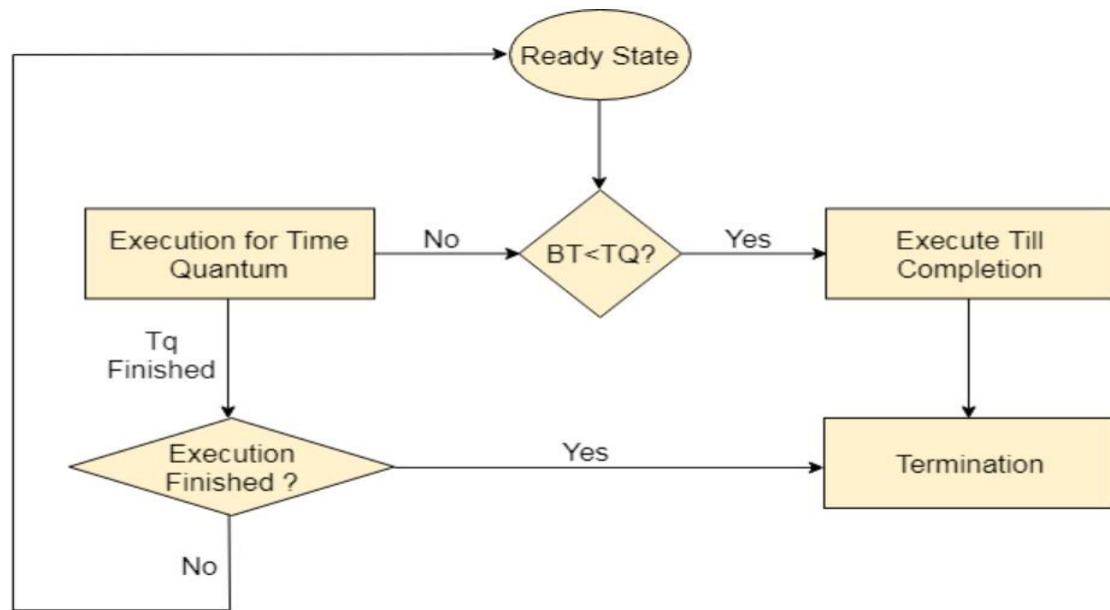
9. Go Live:

Once testing is successful and you're confident in the integration, switch from the test environment to the live production environment to start processing real transactions.

10. Maintenance and Updates:

Regularly update and maintain the integration to ensure compatibility with any updates from the payment gateway provider and to address security vulnerabilities.

Each payment gateway provider may have its own specific integration process and documentation. Refer to the provider's official documentation and developer resources for detailed integration guidelines and support.



5. CONCLUSION:

The main purpose of the project is to develop a useful product which is an android application for the Electric Vehicle users which will be very convenient for them. This app will not only provide service to the user, but it will also be used by the vendor which is Electric station owners they act as admin as an interactive system. It can also generate more data of user which owns the electric vehicles and also the vendors which have chargingstation. By using this one can used to find as well as to navigate to stations. This app will also be expanded in the future as a commercial product with more features that will also use subscription packs and as well as features like charge and chill which will generate more revenue.

- K-NN is a fundamental machine learning algorithm, offering simplicity and effectiveness in handling classification and regression tasks. However, it requires careful consideration of parameters and may not perform optimally with large or high-dimensional datasets.

- Round Robin Scheduling is a fair CPU scheduling algorithm suitable for time-sharing systems but can face performance challenges due to frequent context switches and its sensitivity to time quantum selection.

Both algorithms have their strengths and limitations, and their effectiveness depends on the context of application, dataset characteristics, and system requirements. Understanding their principles and considering their nuances is crucial in deploying them effectively in real-world scenarios.

REFERENCES:

1. Syed Muhammad Danish, Kaiwen Zhang, Hans-Arno Jacobsen, Nouman Ashraf, and Hassaan Khaliq Qureshi, "BlockEV: Efficient and Secure Charging Station Selection for Electric Vehicles", IEEE Transactions on Intelligent Transportation Systems Vol. 22, Issue: 7, July 2020, pp: 4194- 4211, doi: 10.1109/ TITS.2020 .3044890.
2. Suresh Chavhan, Nikhil Dubey , Abhinav Lal , Dev Khetan , Deepak Gupta , Ashish Khanna , Joel J. P. C. Rodrigues, Plácido Rogerio Pinheiro, "Next-Generation Smart Electric Vehicles Cyber Physical System for Charging Slots Booking in Charging Stations ", IEEE Access, Vol. 8, pp: 160145 - 160157, 2020, doi: 10.1109/ACCESS.2020.3020115.
3. Shivi Sharma, Aditya Mathur, Kavita Choudhary, Pushpendra Singh, "Block chain enabled electric vehicle charging infrastructure", IEEE 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), Date of Conference: 15-17 July 2020, doi: 10.1109//ICIRCA 48905 .2020.9183298
4. Jyoti M. Kharade, Mangesh P. Gaikwad, Saurabh P. Jadhav, Parag D. Kodag, Sweta P. Pawar, Supriya

T. Yadav, "IOT based charging slot locator at charging station", IEEE 2020 5th International Conference on Communication and Electronics Systems (ICCES), Date of Conference: 10-12 June 2020, doi: 10.1109/ICCES48766.20209137937

5. https://www.google.com/search?q=AI-powered+EV+charging+station+locator+and+slot+booking+system&sca_esv=590033983&tbm=isch&sxsrf=AM9HkKkKrYVQKibTSeVQY2b1ZdZbIK2dgg:1702356390109&source=lnms&sa=X&ved=2ahUKEwiLjKffi4mDAxW8yDgGHe9NBLUQ_AUoAnoECAEQBA&biw=1366&bih=633&dpr=1#imgrc=u2QvWe8wlHjudM

6. Zhiyong Tian, Taeho Jung, Yi Wang, Fan Zhang, Lai Tu, Chengzhong Xu, Chen Tian, and Xiang-Yang Li, "Real-time Charging Station Recommendation System for Electric-Vehicle Taxis", IEEE Transaction on Intelligent Transportation systems Volume 17, Issue 11 November 2018, doi: 10.1109/TITS.2018.2539201

7. "A Comparison of CPU Scheduling Algorithms for Real-Time Systems" by T.L. Shu and M.T. Liu. (IEEE Xplore, ACM Digital Library, or similar academic databases.)

8. "Pattern Recognition and Machine Learning" by Christopher M. Bishop.

9. "Introduction to Statistical Learning" by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani.

10. "Operating System Concepts" by Abraham Silberschatz, Peter B. Galvin, and Greg Gagne.

11. "Modern Operating Systems" by Andrew S. Tanenbaum and Herbert Bos.

