

# ELECTRIC VEHICLE CHARGING STATION RECOMMENDATIO...

By: Abhinna Ojha

As of: May 3, 2023 11:22:29 AM  
4,827 words - 34 matches - 21 sources

Similarity Index

7%

Mode: Similarity Report ▼

paper text:

Tribhuvan University Orchid International College A Final Year Project Report On

1

ELECTRIC VEHICLE CHARGING STATION RECOMMENDATION SYSTEM USING

COLLABORATIVE FILTERING Under the Supervision of Er . Utsab Koirala Lecturer Orchid  
International College Submitted To

1

:

Orchid International College In partial fulfillment of the requirement for the Bachelor Degree in Computer  
Science and Information Technology Submitted By

1

: Rajat Budhathoki (20820/075) Bipin Chaudhary (20800/075) Abhinna Ojha (20788/075) April, 2023

CHAPTER 1: INTRODUCTION 1.1. Introduction Electric vehicles

19

, EVs, are gaining popularity in recent years. The reduction of taxes by various nations on EVs, the consideration of environment, and the ever-depleting petroleum resources has made EVs a possible vehicle in consideration for many drivers and the easier and cheaper availability of electricity further solidifies the case for the choice of EVs. Similarly, recommendation systems are also nowadays used on almost every walk of digital life. The increasing adoption of electric vehicles (EVs) has led to a growing demand for charging infrastructure. However, finding a charging station that is both conveniently located and available can be a challenge for EV drivers. The aim of this project is to address this challenge by providing a user-friendly and efficient solution for recommending charging stations. The system utilizes the technique of cosine similarity to recommend the most suitable electric vehicle charging stations in a specified location. The report will

detail the design, implementation and evaluation of the system, as well as its overall performance and potential future improvements.

**1.2. Problem Statement** The big problem of Electric Vehicles (EV), is charging. EV charging is not as easy as filling up fuel as in an internal combustion engine-based vehicle as it takes quite a significant time for the EV to be fully charged. So, picking an EV charging station can be a big personal preference-based choice and finding a right one can be a topic of choice. With the scarce availability of the EV charging stations, it can be very difficult to even look for an EV charging station let alone being recommended one that a person may like, trust, and ultimately prefer.

**1.3. Objectives ?** To create a web-based recommendation system that recommends EV charging stations in an area ? To implement item-based collaborative filtering using cosine similarity and weighted average

**1.4. Scope and Limitation** The project aims to build a system that can be used to provide ratings to various charging stations and receive recommendation of charging station in a user specified location. The system will not detect the distance to the nearest amenities or detect the geo-code during insertion of charging station.

**1.5. Development Methodology** The project is developed using incremental development approach. Incremental development is a method of building software where the system is broken down into small modules and built module-by-module. Incremental development was favoured over other approaches due to multitude of factors. The team is not very well skilled or trained and is on a constant learning process during the development process itself. This approach allowed so that the system can be developed in smaller modules and functions can be added over the course of development one by one. Similarly, the system could be developed such that the core requirements of the system could be developed first and then other add-ons could be developed as need be. Furthermore, due to the requirements being set and clear, it was easier to implement the system in incremental approach as client feedback was not a must have requirement for the project. All in all, as requirements are superior, the developers are still learning, and the prioritized requirements need to be developed first, the system is built using incremental approach as small modules.

**1.6. Report Organisation** The report of the Electric Vehicle Charging Station Recommendation System is organised into six chapters. Chapter 1 is the introductory chapter that encompasses the project introduction, the scope, limitation, and development methodology. Chapter 2 is the background study and the literature review. This section encompasses the summary of the works done which are similar to the project in consideration. It is research-based analysis and summary of similar works or works that may be relevant to the project. This section summarises the works that have been taken as reference, guideline, or as an influence to the project. Chapter 3 provides a detailed analysis of the system and the project as a whole. This section focuses on identifying the viability of the project and the requirement of the 2 system. This section ensures that the system is viable, the platform to be used is viable, and the project is doable. Chapter 4 highlights the system and how is it to be developed. It is a study of how should the system be built, which approach is to be taken, and what algorithms are to be studied and implemented to ensure smooth and hassle-free development along with providing a system of proper quality. Chapter 5 provides an outlook on how was the system actually developed and how was it handled. It provides summary on what tools were implemented and why, how was the algorithm implemented, and how was the database implemented. It also summarises the QA aspects of the project. Finally, chapter 6 provides an insight about the system and the project and how can the system be further improved.

**CHAPTER2: BACKGROUND STUDY AND LITERATURE REVIEW**

**2.1. Background Study** Electric Vehicle is not a new concept. It has been a topic of discussion and research as an alternative fuel vehicle since the beginning of 21st century. With the breakthrough in research in recent years, and a global awareness of the adverse climatic effects due to fossil fuels, electric vehicles has been a prominent and upcoming field of research. The

biggest problem of the EV is the travel distance as the per charge range is significantly lower than the per tank range of ICE based vehicles. The other fuel system is the use of Hydrogen fuel, but the problematic storage of Hydrogen and Oxygen required for the fuel cell makes the process of fuel cell-based vehicles a challenging concept. Solar power can be used, but solar power produces inconsistent power and may be subject to higher cost due to the innate costs of the panels, maintenance, and repair. So, with careful consideration to all the possibilities, EVs with L-ION batteries and accessible charging stations seems to be the most likely future for the next few decades. Research in EVs have been prominent and research into ways to increase the range of EVs have been a topic of heavy interest. But, for now, it seems though that the availability of suitable EVCS is the better approach towards EV charging. Furthermore, most of the EVCS recommendation system is based on technicalities. EV charging is treated as ICE vehicle fuelling and recommendations are based on technical parameters only. While technical parameters are important and effective, EV charging is more than just the technicalities. EV charging is not as quick and robust as ICE fuelling and the human aspect is mostly neglected by most researches and systems. While recommendation systems like that of movies is based on human preferences, the recommendation system of EVCS mostly does not consider potential human factors like what will they do during the downtime when their vehicle is charging and how can recommendation be made more human-centric.

### 2.2. Literature Review

Many prominent researches have been done on the recommendation systems and few of them are based on the electric vehicle charging station. In a study on Electric Vehicles in Hyderabad, the Urooj and Annamma research on the EV taxies in Hyderabad, and how can the EVCS recommendation be beneficial to the EV taxies. The EV taxies were compared based on real and simulated recommendations with an accuracy of

94.7%, among which the accuracy of to-station state prediction is 92.2%, 96.8% for recharging, and

2

95.3% for operating. The comparison of real selections with predicted selections was obtained as 84.7% on

integral recharge intention identification, including to-station state, and charging station selection

2

[1]. Research in a similar system was conducted in 2021 by Zhang et al which discusses recommendation system of EVCS. The research focused on various parameters like charging request, charging wait time, charging price, and charging failure rate to recommend the most proper charging station with long term goals of minimising the charging wait time, charging price, and charging failure rate[2]. Aarthi and Prasath proposed an enhanced

real-time charging station recommendation system for the EV

2

charging station recommendation system for load-based taxis. The

13

study suggested

that EV taxis ' higher time-cost at the charging stations is due to the EV taxi drivers rushing to the same EV charging stations during the same

2

time-frame. The system also suggested the use of

real-time recommendation system so that the taxi drivers

13

can get real time suggestions so that they do not rush only to the charging stations they frequently use[3]. The study by Wang et al in Beijing in 2021 also proposed a similar recommendation system, albeit using federated learning. The research compared the model based on real data and that based on their model and concluded that the model was accurate by a factor of 0.92, 92%[4].

## CHAPTER 3: SYSTEM ANALYSIS 3.1. Requirement Analysis

20

Requirements analysis is the process of obtaining what the client, user, or stakeholders expect from the system or software. The functional requirements specify the core features and the expected behaviours of the system and is modelled using Data Flow Diagrams, UML diagrams, flowcharts, function modelling, and so on. The non-functional requirements are

a set of specifications that describe the system's operation capabilities and constraints and attempt to improve its functionality

7

3.1 .1. Functional Requirements The functional requirements specify the core expected behaviour of the system

6

. It

is a description of the service that the software must offer and describes a software system or its component

4

. In this system some core functions that have been modelled sing the use- case diagram and use-case descriptions. The core features of the system are to add and update charging stations by the admin, add and edit charging station ratings by the user, and get recommendation of charging station by user.

3.1.1 .1. Use-case diagram Figure 3.1: Use case of the System 3.1.1

12

.2. Use-case descriptions Table 3.1: Use case description of rating a charging station Identifier UCD1 Name, Description Providing the rating to a charging station Actors User Pre-condition The location must be specified and charging station must be selected Success scenario Rating is insertedinto the database Success message is displayed to user Failure scenario Rating is not inserted into the database and failure message is displayed to admin Table 3.2: Use case description of adding a charging station Identifier UCD2 Name, Description Adding a charging station Actors Admin Pre-condition The location must be specified Success scenario Charging station is inserted into the database Success message is displayed to admin Failure scenario Charging station is not inserted into the database and failure message is displayed to admin Table 3.3: Use case description of updating a charging station Identifier UCD3 Name, Description Updating a charging station Actors Admin Pre-condition The location must be specified and the charging station to update must be selected Success scenario Success message is displayed to admin Charging station is updated in database Failure scenario Charging station is not updated in database and message is displayed to admin Table 3.4: Use case description of getting a charging station recommended Identifier UCD4 Name, Description Get recommendation of charging station Actors User Pre-condition The location must be specified Success scenario Charging stations and their location is displayed to user if the charging station is available Message is displayed to user if the location has no charging station Failure scenario Charging station is not recommended as desired 3.1.2. Non-Functional Requirements ? Speed: It is used to determine how fast the system performs certain activities. ? Security: Only registered users can login to the system and provide ratings or get recommendations. ? Availability: The system will be available at all hours, every day of the year once deployed. ? Usability: The system should be easy to access, use, and understand by the users 3.2. Feasibility Study 3.2.1. TechnicalFeasibility The dataset to train and test the recommendation system can be available from sources like NEA and Kaggle. Laravel framework would be used to build the recommendation system and the collaborative filtering technique with cosine similarity would be used for the recommendation algorithm. The next step would be to develop the recommendation system using the Laravel framework and the chosen algorithm. This would involve the implementation of various features such as user authentication, data management, and recommendation generation. The developed system would then be tested to ensure its functionality and performance. This would involve evaluating the system's accuracy and efficiency in generating

recommendations, as well as user satisfaction with the system. Once the system has been tested and evaluated, it would be deployed for use by EV owners. This would involve the integration of the system with existing EV charging stations and the implementation of any necessary security measures. All in all, the technical feasibility of this project is high as the Laravel framework is a robust and widely used platform for web development, and collaborative filtering with cosine similarity is a well-established recommendation algorithm. With the appropriate data and resources, the development, testing, and deployment of this system is achievable.

**3.2.2. Operational Feasibility** The system will be able to recommend the EV charging station to the user in the specified location. This recommendation will be based on the ratings the user has provided to the EV charging stations that the user has already been to and rated. This solves the problem for the user to manually search the EV charging station and make a guess of whether the one they found may be one of their liking or not and whether they will be able to trust their expensive vehicles on the said charging station or not. Furthermore, having a system of authentication and authorisation will also help prevent fake ratings and make the system more reliable and robust.

**3.2.3. Schedule Feasibility**

<b>Figure 3.2: Work breakdown structure</b>	<b>Figure 3.3</b>	17
---	-------------------	----

: Gantt chart

<b>3.3. System Analysis</b>	<b>3.3.1 . Class Diagram</b>	<b>Figure 3 .4: Class diagram</b>	15
-----------------------------	------------------------------	-----------------------------------	----

<b>3.3.2 . Activity Diagram</b>	<b>Figure 3.5 : Activity diagram</b>	8
---------------------------------	--------------------------------------	---

to provide rating or receive recommendation

<b>3.3.3. Sequence Diagram</b>	<b>Figure 3.6: Sequence diagram</b>	8
--------------------------------	-------------------------------------	---

to receive recommendation after adding a station

**CHAPTER 4: SYSTEM DESIGN**

**4.1. Design**

**4.1.1. Model View Controller Architecture** The system is built following the Model View Controller, MVC, architecture. The model handles the data logic, database communication, and data on database. The controller contains the core business and implementation logic. It also acts as the intermediary between the model and view and helps render views via routes. The view renders the pages and acts as the presentation part of this system. As MVC is one of the most popular and efficient design architectures for web-based systems, we chose to use MVC architecture for this system.

**Figure 4.1: Model View Controller architecture**

**4.2. Study of Algorithms** Collaborative filtering is a technique

**to provide** suggestions **to the users based on**

10

their preferences and reviews. Collaborative filtering can broadly be implemented by two approaches; the

**user-based approach and** the **item-based approach. The user-based approach** uses data from  
the

14

various users to find similarities and then recommend while the item-based approach uses data from the various items to find similarities and then recommend. Furthermore, collaborative filtering can be implemented using memory-based, model-based, hybrid, or deep learning methods. Memory-based approach uses similarity calculation and weighted average rating method. Model based approach uses model-based data mining and/or machine learning algorithms to compute and predict the missing ratings of items not rated by users. The hybrid approach combines the

**memory-based and** the **model-based CF** algorithms. **The**

21

hybrid approach overcomes the limitations of native Collaborative Filtering approaches and improve performance of the predictions. Deep learning technique is a recent method that uses neural and deep-learning techniques. While deep learning has been applied to many different scenarios, it is not effective when used in a simple collaborative recommendation scenario. 4.2.1.

**Item-based Collaborative Filtering The system**

5

utilises

**an item based collaborative filtering algorithm** . Rather than matching **the**

5

user to other

**similar users, item based collaborative filtering** matches each of **the**

5

user's rated items to other similar items, then recommends those similar items as a recommendation list. Cosine similarity and

Pearson correlation are the most commonly used method to calculate the10

similarity scores used for the collaborative filtering process. 4.2.2. Memory-based Approach Memory-based approach calculates the similarity between two entities, and predicts the missing rating

for the user by taking the weighted average of all the ratings provided by the user or provided to the item9

.Similarity measures like

Pearson correlation and vector cosine-based similarity are used for16

this.Despite of its problems with sparse data, it is a highly effective method due to the results being explainable, method itself being easy to use, new data being easier to facilitate into the system, and having good scalability with co-related items. 4.2.3. Cosine Similarity Cosine similarity measures the similarity by computing the Euclidean dot product between the two vectorial data. The cosine similarity calculation generates an outcomebelonging to the interval [-1, 1]. The cosine of two non-zero vectors, X and Y, can be derived by using the formula:  $W = \frac{X \cdot Y}{\|X\| \|Y\|}$ . If X and Y are two n-dimensional vectors,the cosine similarity, is represented using a dot product and magnitude as:  $Rc(W, W) = \cos(\theta) = \frac{W \cdot W}{\sqrt{\sum_{i=1}^n W_i^2} \cdot \sqrt{\sum_{i=1}^n W_i^2}}$ .  $W = \sum_{i=1}^n W_i$  where Xi and Yi are components or attributes of vector X and Y respectively. 4.2.4. Weighted Average Weighted average is a computation of an average that takes into account the importance of various data in the set. Each data in the data set is multiplied by a factor or degree of importance, which is the weight, before the final calculation is made.  $\sum_{i=1}^n w_i W_i$   $W = \sum_{i=1}^n w_i$  CHAPTER 5: IMPLEMENTATION 5.1. Tools Used 5.1.1. Development Tools 5.1.1.1. HTML and CSS Both HTML and CSS is used this project to create the basic structure and the user interface components in the views. 5.1.1.2. JS along with jQuery This project uses JS and jQuery to provide interactivity and visual appeal of the web pages as well as making asynchronous calls to the database. 5.1.1.3. PHP with Laravel In his project PHP with Laravel framework is used to implement the back-end logic and the ORM structurers. It is the base of the project that handles all the core logic and algorithms as well as providing a MVC structure for development. The system is built using PHP 8.0 and Laravel 9.47.0. 5.1.1.4. PHP Storm IDE The system is developed using PHP Storm IDE with student license. It was chosen as it provides a better, leaner, and easier development interface and a better IntelliSense compared to other alternatives. The availability of free student license allowed for the project cost to be reduced. 5.1.1.5. MySQL The core database of the project is built in MySQL using PHP as PHP offers an easy integration of the MySQL databases. 5.1.1.6.



GitHub with Git Git is a version control system. And GitHub is an online tool that uses git for version control and team collaboration. The combination of these tools was used for project versioning and code collaboration within the project team. 5.1.1.7. XAMPP server In this project the XAMPP server allowed the simulation of the Apache server to run the PHP scripts and implement the MySQL database. 5.1.2. Design and Documentation Tools 5.1.2.1. diagrams.net The project uses diagrams.net to prepare the

various UML diagrams like use-case diagram and activity diagram

18

. 5.1.2.2. Microsoft Project In this project Microsoft Project is used as the project analysis tool that allowed for the schedule analysis and planning using work breakdown structure and Gantt charts. 5.1.2.3. Microsoft Word Microsoft Word is a word processor software. It is used during the report and proposal preparation for the project. 5.1.2.4. Microsoft PowerPoint In this project, Microsoft PowerPoint is used for preparing the presentation slides during the proposal submission and defence, mid-term defence, and final defence. 5.2. Database Implementation Figure 5.1: Database implementation 5.3. Algorithm Implementation This system uses item-based approach and a memory-based method to implement collaborative filtering. The system implements vector cosine-based similarity to compute the similarity score required for the memory-based approach. The algorithm is implemented in two phases. The first phase is the computation of similarity scores between two charging stations while the second phase is the execution of the recommendation system using weighted average. 5.3.1. Phase 1: Similarity between charging stations using cosine similarity The first phase in the implementation of algorithm in the system is the implementation of cosine similarity between the charging stations. Unlike the general approach of using the user ratings itself for the computation of similarity scores to generate a similarity matrix, this system used the attributes of the charging stations itself to factor into the computation of the similarity matrix. The system calculates the similarity scores using the cosine similarity as mentioned in section 4.2.3 of this document. 
$$\text{sim}(A, B) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$
 Here, A and B are the two charging stations vectors whose similarity scores are to be computed and  $\text{sim}$  is the similarity score obtained. The attributes of the charging stations that contribute to the computation of similarity scorers are: ? Number of Type 1 AC charging ports, AC regular charging ports ? Number of Type 2 AC charging ports, AC fast charging ports ? Number of CHAdeMO charging ports, DC regular charging ports ? Number of Combined Charging System (CCS) charging ports, DC fast charging ports ? Distance from the nearest restaurant ? Distance from the nearest shopping mall ? Distance from the nearest cinema hall The distances are first transformed into discrete values using a discretisation method. The distances are discretised based on the distance value. The discretisation transformation method is implemented so that the large distance values can be transformed into smaller values so that it is easier to compute. The distance is not as sensitive so the discretisation induced data loss is not a major issue. The discretisation is done as: Table 5.1: Discretisation of distance S.N. Distance Discretised 1 >500 0 2 >450 and <=500 1 3 >400 and <=450 2 4 >350 and <=400 3 5 >300 and <=350 4 6 >250 and <=300 5 7 >200 and <=250 6 8 >150 and <=200 7 9 >100 and <=150 8 10 >50 and <=100 9 11 >0 and <=50 10 12 =0 0 The dot product of the two vectors is then calculated using the above-mentioned formula. There are 7 attributes involved in the similarity calculation, so taking  $n = 7$ , the formula can be re-

written with consideration of the attributes as:  $sim(A, B) = \frac{\sum_{i=1}^7 a_i b_i}{\sqrt{\sum_{i=1}^7 a_i^2} \sqrt{\sum_{i=1}^7 b_i^2}}$  If the 7 attributes of  $A$  are named as

$a_1, a_2, a_3, a_4, a_5, a_6$  , and  $a_7$  and that of  $B$  are named as  $a_1, a_2, a_3, a_4, a_5, a_6$  , and  $a_7$  3

then the similarity score is computed as:  $Rc(A, B) =$

$a_1.a_1 + a_2.a_2 + a_3.a_3 + a_4.a_4 + a_5.a_5 + a_6.a_6 + a_7.a_7$  3

$\sqrt{\phantom{x}}$

$a_1^2 + a_2^2 + a_3^2 + a_4^2 + a_5^2 + a_6^2 + a_7^2$  11

$\cdot \sqrt{\phantom{x}}$

$a_1^2 + a_2^2 + a_3^2 + a_4^2 + a_5^2 + a_6^2 + a_7^2$  11

5.3.2. Phase 2: Predicted rating generation using weighted average The second phase is the implementation of the predicted rating calculation to find which charging station is to be recommended. The system calculates the weighted average using the method as mentioned in section 4.2.4 of this document.  $\sum_{i=1}^n w_i W_i$  Here,  $w_i$  is the weight value,  $W_i$  is the item value and  $W$  is the weight obtained after the computation. For this system, the weight is the similarity score between the charging stations. The item value is the rating provided to the charging station by the user. So, if we consider  $P(a, u)$  as the predicted rating of charging station  $a$  for user  $u$ ,  $R_{c,ci}$  as the similarity score between the charging station  $a$  and  $a_i$ , and  $R_{u,ci}$  as the rating provided by user  $u$  to charging station  $a_i$  then we can rewrite the formula as:  $\sum_{i=1}^n R_{c,ci} R_{u,ci}$

5.4. Testing Testing is the process subjecting the system to various conditions, scenarios, or constraints so as to find whether the system performs correctly and desirably. Testing is performed parallelly to coding. If any bugs or errors are found during the testing phase, it is fixed via editing the code or re-coding. Validation ensures that the software satisfies the user requirements. If the software matches the user requirements, it is said to be validated. Verification makes sure that the system is developed following the proper specifications and methodologies. If the software matches the development criteria it is said to be verified. This system was subject to various scenarios and the outcome was noted. The system was tested using test cases which have been documented as: Table 5.2: Test case for signup Test Case ID TC-1 Test Scenario Register new user Actions 1. Open Register page 2. Input the name, email, and password. 3. Click

“Register” button. Input Name: Ram Dhami Email: ramdhami@gmail.com Password: 23morang@33 Confirm password: 23morang@33 Expected Results ? Redirected to recommendations page. ? Ram Dhami is displayed on top right of screen. Observed Results As expected Assertion Pass Table 5.3:

<b>Test case</b> for <b>login Test Case ID TC-2 Test Scenario Login</b>	<b>6</b>
---	----------

existing user Actions 1. Open Login page 2. Input the email and password. 3. Click “Login” button. Input Email: ramdhami@gmail.com Password: 23morang@33 Expected Results ? Redirected to recommendations page. ? Ram Dhami is displayed on top right of screen. Observed Results As expected Assertion Pass Table 5.4: Test case for addition of rating with valid data Test Case ID TC-3 Test Scenario Addition of rating using valid data. Actions 1. Open rate page. 2. Input the location, the charging station, and rating. 3. Press “Add Rating” button. Input Province: Bagmati District: Kathmandu Metropolitan: Kathmandu Ward: 22 Charging Station: Jagat Charging Station Rating: 4 Expected Results ? Success message is displayed. ? “ratings” table is updated. Observed Results As expected Assertion Pass Table 5.5: Test case for addition of charging station with valid data Test Case ID TC-4 Test Scenario Addition of charging station with valid data. Actions 1. Open Add Charging Station page. 2. Input the locations and the charging station details. 3. Press “Add Charging Station” Input Name: Province: District: Metropolitan: Ward: Fast Charging AC Ports: Fast Charging DC Ports: Regular AC Ports: Regular DC Ports: Nearest Restaurant: Nearest Shopping Mall: Nearest Cinema Hall: Shiva Shakti Charging Station Bagmati Kathmandu Kathmandu 30 2 3 4 4 312 569 452 Expected Results ? Success message is displayed ? “charging\_stations” table has a new entry with the columns having the values as inputted.s ? Redirect to index page. Observed Results As expected Assertion Pass Table 5.6: Test case for addition of charging station with invalid data Test Case ID TC-5 Test Scenario Addition of charging station using invalid data. Actions 1. Open Add Charging Station page. 2. Input the locations and the charging station details. 3. Press “Add Charging Station” button. Input Name: Province: District: Metropolitan: Ward: Fast Charging AC Ports: Fast Charging DC Ports: Regular AC Ports: Regular DC Ports: Nearest Restaurant: Nearest Shopping Mall: Nearest Cinema Hall: Jagat Charging Station Bagmati Kathmandu Kathmandu 22 2 3 4 4 312 569 -1 Expected Results ? System does not allow the invalid data and error message is displayed Observed Results As expected Assertion Pass Table 5.7: Test case for getting recommendation without ward Test Case ID TC-6 Test Scenario Get recommendation of charging station without inputting ward. Actions 1. Open Recommendations page. 2. Input the location and check the Exclude Ward checkbox. 3. Press “Get recommendation” button. Input Province: Bagmati District: Kathmandu Metropolitan: Kathmandu Expected Results ? Top 3 charging stations displayed. Observed Results As expected Assertion Pass Table 5.8: Test case for getting recommendation with ward Test Case ID TC-7 Test Scenario Get recommendation of charging station. Actions 1. Open Recommendations page. 2. Uncheck the Exclude Ward checkbox. 3. Input the location. 4. Press “Get recommendation” button. Input Province: Bagmati District: Kathmandu Metropolitan: Kathmandu Ward: 3 Expected Results ? Top 3 charging stations displayed. Observed Results As expected Assertion Pass

CHAPTER 6: CONCLUSION AND FUTURE RECOMMENDATIONS 6.1. Conclusion The product of this work, Electric Vehicle Charging Station Recommendation System, is a web-based application that provides a platform that recommends EV charging stations to the user in their desired location. The memory-based approach of cosine similarity and weighted average allows the system to

have decent scalability on co-related items. The system has functional authentication, similarity calculation during insertion, rating, and recommendation. The system is able to recommend the charging station in the said location and the algorithm functions as intended. Existing recommendation systems had a problem of specialisation. They were either too technical and thus overlooked the human aspect as EV charging is not a quick endeavour, or overlooked the technical aspect and were recommending based on ratings of users alone, which in times recommended incompatible charging stations. The system incorporates both the technical and humanaspects of the charging station to provide a better recommendation for the users.

6.2. Future Recommendations In spite of the system having good performance, the system could still be improved upon. The system can incorporate maps and use dynamic distance computation to provide a more dynamic system to the users as well as the admin. PHP is not well suited for implementation of high-end ML and AI algorithms. A change in the choice of the backend framework could allow implementation of a hybrid filtering approach, despite being highly complex. Finally, in the era of real-time technology, the system could be subject to further improvements by adding real-time booking and scheduling of the EV charging station ports.

1 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23  
24 25 26 27 28

**sources:**

1

45 words / 1% - from 25-Apr-2023 12:00AM  
[WWW.coursehero.com](http://WWW.coursehero.com)

2

59 words / 1% - Internet from 11-Mar-2022 12:00AM  
[autodocbox.com](http://autodocbox.com)

3

29 words / 1% - Internet from 29-Oct-2013 12:00AM  
[www.boost.org](http://www.boost.org)

4

18 words / < 1% match - Internet from 27-Mar-2022 12:00AM  
[www.coursehero.com](http://www.coursehero.com)

5

20 words / < 1% match - ProQuest  
[Somani, Chetan Prakash. "Item-Based Recommendation Algorithm Using Hadoop", Auburn University, 2023](#)

6

20 words / < 1% match - Internet from 05-Jun-2012 12:00AM  
[publications.theseus.fi](http://publications.theseus.fi)

7

19 words / < 1% match - Internet from 24-Dec-2022 12:00AM  
[github.com](http://github.com)

8

18 words / < 1% match - Internet from 11-Mar-2021 12:00AM  
[www.mgit.ac.in](http://www.mgit.ac.in)

- 9 16 words / < 1% match - Internet from 05-Jan-2021 12:00AM  
[www.hindawi.com](http://www.hindawi.com)
- 
- 10 14 words / < 1% match - Crossref  
["Recommender Systems Handbook", Springer Science and Business Media LLC, 2015](#)
- 
- 11 14 words / < 1% match - Crossref  
[Ioannis Antonellis. "Personalized News Categorization Through Scalable Text Classification", Lecture Notes in Computer Science, 2006](#)
- 
- 12 14 words / < 1% match - Internet from 23-Sep-2022 12:00AM  
[eprints.utar.edu.my](http://eprints.utar.edu.my)
- 
- 13 13 words / < 1% match - Crossref  
[Zhiyong Tian, Taeho Jung, Yi Wang, Fan Zhang, Lai Tu, Chengzhong Xu, Chen Tian, Xiang-Yang Li. "Real-Time Charging Station Recommendation System for Electric-Vehicle Taxis", IEEE Transactions on Intelligent Transportation Systems, 2016](#)
- 
- 14 12 words / < 1% match - Internet from 26-Dec-2022 12:00AM  
[deepai.org](http://deepai.org)
- 
- 15 11 words / < 1% match - Internet from 14-Nov-2022 12:00AM  
[repository.sustech.edu](http://repository.sustech.edu)
- 
- 16 10 words / < 1% match - Crossref  
[Kannan, S., S. Karuppusamy, A. Nedunchezian, P. Venkateshan, P. Wang, N. Bojja, and A. Kejariwal. "Big Data Analytics for Social Media", Big Data, 2016.](#)
- 
- 17 9 words / < 1% match - Internet from 05-Mar-2023 12:00AM  
[utpedia.utp.edu.my](http://utpedia.utp.edu.my)
- 
- 18 8 words / < 1% match - from 07-Apr-2023 12:00AM  
[cmuir.cmu.ac.th](http://cmuir.cmu.ac.th)
- 
- 19 8 words / < 1% match - Internet from 27-Jan-2023 12:00AM  
[fr.slideshare.net](http://fr.slideshare.net)
- 
- 20 8 words / < 1% match - Internet from 11-Jun-2010 12:00AM  
[www.seminarprojects.com](http://www.seminarprojects.com)
- 
- 21 7 words / < 1% match - Crossref  
[Feng Xia, Nana Yaw Asabere, Ahmedin Mohammed Ahmed, Jing Li, Xiangjie Kong. "Mobile Multimedia Recommendation in Smart Communities: A Survey", IEEE Access, 2013](#)

