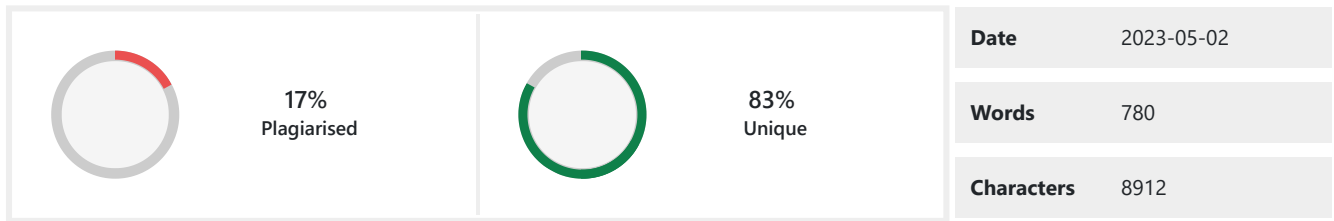


## PLAGIARISM SCAN REPORT



## Content Checked For Plagiarism

### CHAPTER 6: CONCLUSION AND FUTURE RECOMMENDATIONS

1.

#### 1.1. 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 human aspects of the charging station to provide a better recommendation for the users.

#### 1.2. 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.

### REFERENCES AND BIBLIOGRAPHY

[1]

R. Urooj and K.

**Annamma, "Persistent Charging Station Recommendation System for Electric-Vehicle Taxis," International Journal of Scientific Engineering and Technology Research, Hyderabad, India, 2017.**

[2]

W. Zhang, H. Liu, F. Wang, T. Xu, H. Xin, D. Dou and H.

**Xiong, "Intelligent Electric Vehicle Charging Recommendation Based on Multi-Agent Reinforcement Learning," International World Wide Web Conference Committee, Ljubljana, Slovenia, 2021.**

[3]

R.Aarthi and P. Prasath, "Enhanced Real-Time Charging Station Recommendation System For Load Base Electric-Vehicle Taxis," South Asian Journal of Engineering and Technology, 2017.

[4]

X. Wang, X. Zheng and X. Liang, "Charging Station Recommendation for Electric Vehicle Based on Federated Learning," Artificial Intelligence on Electric Power System State Grid Corporation Joint Laboratory, Beijing, China, 2021.

[5]

F. Ricci, L. Rokach and B. Shapira, **Introduction to Recommender Systems Handbook**, 2011.

[6]

X. Su and T. M. Khoshgoftaar, **"A Survey of Collaborative Filtering Techniques,"** 2009.

[7]

A. Singhal, "Modern Information Retrieval: A Brief Overview," Google, Inc..

## APPENDIX

Snippets of major source code components

Computation of similarity scores

```
function calculateSimilarityScores($cs1, $cs2) {
    $chargingStationModel = new ChargingStation();
    $cs_att_1 = $chargingStationModel->getChargingStationAttributes($cs1);
    $cs_att_2 = $chargingStationModel->getChargingStationAttributes($cs2);
    $ab = $cs_att_1[0]->ac_ports_fast * $cs_att_2[0]->ac_ports_fast +
    $cs_att_1[0]->dc_ports_fast * $cs_att_2[0]->dc_ports_fast +
    $cs_att_1[0]->ac_ports_regular * $cs_att_2[0]->ac_ports_regular +
    $cs_att_1[0]->dc_ports_regular * $cs_att_2[0]->dc_ports_regular +
    $this->distance_scale($cs_att_1[0]->nearest_restaurant) *
    $this->distance_scale($cs_att_2[0]->nearest_restaurant) +
    $this->distance_scale($cs_att_1[0]->nearest_shopping_mall) *
    $this->distance_scale($cs_att_2[0]->nearest_shopping_mall) +
    $this->distance_scale($cs_att_1[0]->nearest_cinema_hall) *
    $this->distance_scale($cs_att_2[0]->nearest_cinema_hall);
    $aSquared = $cs_att_1[0]->ac_ports_fast * $cs_att_1[0]->ac_ports_fast +
    $cs_att_1[0]->dc_ports_fast * $cs_att_1[0]->dc_ports_fast +
    $cs_att_1[0]->ac_ports_regular * $cs_att_1[0]->ac_ports_regular +
    $cs_att_1[0]->dc_ports_regular * $cs_att_1[0]->dc_ports_regular +
    $this->distance_scale($cs_att_1[0]->nearest_restaurant) *
    $this->distance_scale($cs_att_1[0]->nearest_restaurant) +
    $this->distance_scale($cs_att_1[0]->nearest_shopping_mall) *
    $this->distance_scale($cs_att_1[0]->nearest_shopping_mall) +
    $this->distance_scale($cs_att_1[0]->nearest_cinema_hall) *
    $this->distance_scale($cs_att_1[0]->nearest_cinema_hall);
    $bSquared = $cs_att_2[0]->ac_ports_fast * $cs_att_2[0]->ac_ports_fast +
    $cs_att_2[0]->dc_ports_fast * $cs_att_2[0]->dc_ports_fast +
    $cs_att_2[0]->ac_ports_regular * $cs_att_2[0]->ac_ports_regular +
    $cs_att_2[0]->dc_ports_regular * $cs_att_2[0]->dc_ports_regular +
    $this->distance_scale($cs_att_2[0]->nearest_restaurant) *
    $this->distance_scale($cs_att_2[0]->nearest_restaurant) +
    $this->distance_scale($cs_att_2[0]->nearest_shopping_mall) *
    $this->distance_scale($cs_att_2[0]->nearest_shopping_mall) +
    $this->distance_scale($cs_att_2[0]->nearest_cinema_hall) *
    $this->distance_scale($cs_att_2[0]->nearest_cinema_hall);
    if ($aSquared == 0 || $bSquared == 0) {
        $similarityScore = 0;
    } else {
        $similarityScore = $ab / (sqrt($aSquared) * sqrt($bSquared));
    }
    return $similarityScore;
}
```

Computation of missing rating and recommendation

```

function getRecommendation(Request $request) {
    $ratingsModel = new Ratings();
    $chargingStationModel = new ChargingStation();
    $recommendationRating = [0, 0, 0];
    $recommendations = [0, 0, 0];
    $user_rating = $ratingsModel->userRatings();
    $chargingStationLocation = '';
    if($user_rating->count() session()->flash('error', 'Please rate at least 3 charging stations.');
```

```

    return redirect()->route('recommendations.index');
} else {
    if($request->get('ward_enabled') == 0) {
        $chargingStationLocation =
        $chargingStationModel->getChargingStationNoWard($request);
    } elseif ($request->get('ward_enabled') == 1) {
        $chargingStationLocation =
        $chargingStationModel->getChargingStationWard($request);
    }
    foreach ($chargingStationLocation as $csl) {
        $ratingEstimateNum = 0;
        $ratingEstimateDen = 0;
        foreach ($user_rating as $ur) {
            $similarityScore = $this->calculateSimilarityScores(
            $ur->charging_station, $csl->charging_station);
            $ratingEstimateNum = $ratingEstimateNum + $ur->rating * $similarityScore;
            $ratingEstimateDen += $similarityScore;
        }
        if ($ratingEstimateDen == 0) {
            $ratingEstimate = 0;
        } else {
            $ratingEstimate = $ratingEstimateNum / $ratingEstimateDen;
        }
        if ($ratingEstimate >= $recommendationRating[0]) {
            $recommendationRating[2] = $recommendationRating[1];
            $recommendationRating[1] = $recommendationRating[0];
            $recommendationRating[0] = $ratingEstimate;
            $recommendations[2] = $recommendations[1];
            $recommendations[1] = $recommendations[0];
            $recommendations[0] = $csl->charging_station;
        } elseif ($ratingEstimate >= $recommendationRating[1]) {
            $recommendationRating[2] = $recommendationRating[1];
            $recommendationRating[1] = $ratingEstimate;
            $recommendations[2] = $recommendations[1];
            $recommendations[1] = $csl->charging_station;
        } elseif ($ratingEstimate >= $recommendationRating[2]) {
            $recommendationRating[2] = $ratingEstimate;
            $recommendations[2] = $csl->charging_station;
        }
    }
}

$provinceModel = new Provinces();
$data['provinces'] = $provinceModel->selectProvinces();
$data['user'] = Auth::id();
$recommendation1 =
$chargingStationModel->getFinalRecommendation($recommendations[0]);

```

```

$recommendation2 =
$chargingStationModel->getFinalRecommendation($recommendations[1]);
$recommendation3 =
$chargingStationModel->getFinalRecommendation($recommendations[2]);
$data['recommendations'] =
collect([$recommendation1, $recommendation2, $recommendation3]);
$data['actual_cs'] = $recommendations;
$data['estimated_rating'] = $recommendationRating;
return view('recommend.recommend', compact('data'));
}

```

Discretisation of Distances

```

function distance_scale($distance_str) {
    $distance = (float)$distance_str;
    if ($distance == 0) {return 0;}
    else if ($distance

```

## Matched Source

### Similarity 7%

**Title:** [Persistent Charging Station Recommendation System for ...](#)

Persistent Charging Station Recommendation System for Electric-Vehicle Taxis. International Journal of Scientific Engineering and Technology Research.

<http://ijsetr.com/uploads/125436IJSETR16037-1206.pdf>

---

### Similarity 7%

**Title:** [Intelligent Electric Vehicle Charging Recommendation Based on ...](#)

WebIntelligent Electric Vehicle Charging Recommendation Based on Multi-Agent Reinforcement Learning WWW '21, April 19–23, 2021, Ljubljana, Slovenia We say a ...

<https://arxiv.org/pdf/2102.07359>

---

### Similarity 6%

**Title:**

[Implementation and evaluation of a resource-based learning ...Guided Walk | Proceedings of the 10th ACM Conference on ...](#)

by M Tahmasebi · 2018 · Cited by 7 — Ricci, F., L. Rokach, and B. Shapira, Introduction to recommender systems handbook. 2011: Springer. Google Scholar Google Scholar ...by R Levin · 2016 · Cited by 17 — F. Ricci, L. Rokach, and B. Shapira. Introduction to recommender systems handbook. 2011. Google Scholar Google Scholar ...

<https://dl.acm.org/doi/10.5555/3370055.3370062>

---

### Similarity 6%

**Title:** [A Survey of Collaborative Filtering Techniques](#)

by X Su · Cited by 4785 — As one of the most successful approaches to building recommender systems, collaborative filtering (CF) uses the known preferences of a group of ...

<https://www.hindawi.com/journals/aai/2009/421425/>

---