Vehicular Edge Computing Schemes Related to Trajectory Prediction

Subjects: Computer Science, Interdisciplinary Applications

Contributor: Yichi Yang , Ruibin Yan , Yijun Gu

As the infrastructure becomes well established, edge servers extend their service coverage to a wider scope. In traffic control, edge computing servers can acquire and regulate real-time traffic. In in-vehicle tasks, edge computing servers can provide high-quality services to users. Vehicular edge computing (VEC) is essential in vehicle applications such as traffic control and in-vehicle services.

edge computing trajectory prediction

1. Introduction

In recent years, research related to edge computing has gradually received extensive attention from researchers [1][2]. Vehicular edge computing (VEC), as a part of edge computing, provides real-time service to vehicular users. It has excellent prospects in the fields of intelligent transportation systems, smart city applications, and vehicular applications.

As the infrastructure becomes well established, edge servers extend their service coverage to a wider scope. In traffic control, edge computing servers can acquire and regulate real-time traffic. In in-vehicle tasks, edge computing servers can provide high-quality services to users. However, the quality of service (QoS) in VEC still cannot be significantly improved, and one of its bottlenecks is the inefficient task offloading. Traditional task offloading methods are plagued by issues such as significant latency, high time and space complexity, and low transmission quality.

To solve the problems of task offloading, trajectory prediction methods are used in the task offloading scheme. For example, tasks which take up a lot of computational resources can be offloaded to other edge servers using predictive-mode multi-hop transmission. Once the vehicle enters the transmission range of the edge server, it obtains the computation results directly [3][4].

2. Vehicular Edge Computing Schemes Related to Trajectory Prediction

2.1. Vehicular Edge Computing

Task offloading in VEC is the process of transmitting the computing task and related parameters from the service requestor to the service providers through Vehicle-To-Vehicle (V2V) and Vehicle-To-Infrastructure (V2I) communications [1]. Saeik et al. [5] summarized the communication issues in task offloading and proposed a novel task offloading scheme that combines edge and cloud resources. An example of vehicular edge computing is shown in **Figure 1** below.

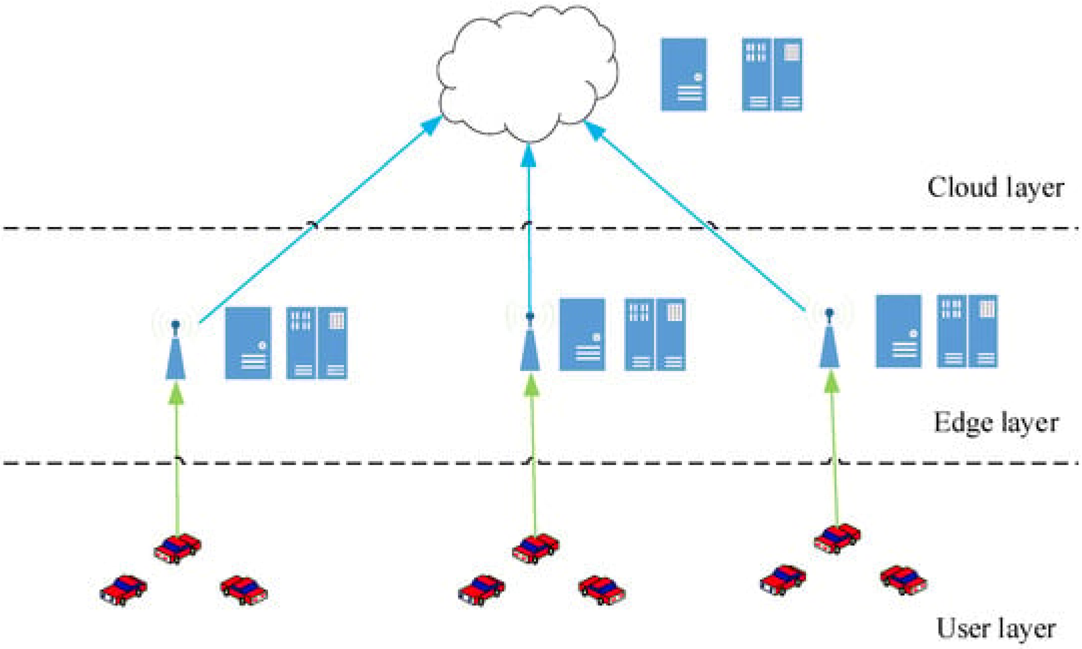

Figure 1. An example of vehicular edge computing.

The resources of edge servers can be fully utilized to provide better QoS to users via optimizing the task offloading scheme. Zhang et al. [4] presented an efficient predictive combination-mode relegation scheme wherein the tasks are adaptively offloaded to the edge servers through direct uploading or predictive relay transmissions. Zhan et al. [3] converted global task offloading optimization problem into multiple local optimization problems with a heuristic mobility-aware offloading algorithm (HMAOA) to approximate the optimal offloading scheme. Yang et al. [6] proposed a low-complexity semiparametric predictive model that takes into account the periodic characteristics and spatial/temporal correlations of dynamic road events. Although these methods have shown some improvements, they still fail to achieve an optimal balance between efficiency and accuracy in VEC. Therefore, how to predict vehicle trajectories more accurately while ensuring efficiency is a pressing issue in task offloading at this stage.

2.2. Trajectory Prediction

Trajectory prediction problems can be categorized into two types based on different data types, namely continuous trajectory prediction problems and discrete trajectory prediction problems. The continuous trajectory prediction problem is a regression problem. Alahi et al. [7] developed an LSTM model which can learn general human movement and predict their future trajectories. Han et al. [8] proposed a short-term real-time trajectory coordinate point prediction method based on a GRU (Gated Recurrent Unit) cyclic neural network. This method improves the accuracy of real-time forecasting by updating the model parameters in real time. Huang et al. [9] discussed a new traffic network modeling algorithm based on the context of traffic intersections that maps vehicle trajectory nodes into a high-dimensional space vector, so that Bi-GRU can be used to bidirectionally model the trajectory matrix for the purpose of prediction. Amichi et al. [10] designed a two-step predictive framework solely based on personal location data. This framework aims to address the prediction of visits to new places and adjust prediction resolution to account for probable explorations of new locations.

Monreale et al. [11] proposed a T-pattern tree for trajectory prediction. The tree is constructed using trajectory patterns that represent specific areas, and it can serve as a predictor for the next location of a new trajectory by identifying the best-matching path within the tree. Dong et al. [12] put forward a new method named RTMatch to predict the future location of a moving object using the storage structure, RTPT and HT, which can be updated dynamically and provide dynamic analysis of trajectory pattern according to real-time information. Zeng et al. [13] presented a next-location prediction approach based on an RNN and self-attention mechanism to predict trajectory patterns based on a sequence of discrete nodes. Feng et al. [14] proposed DeepMove, an attentional recurrent network for mobility prediction from lengthy and sparse trajectories. DeepMove effectively utilizes the periodicity nature to augment the RNN for mobility prediction. Liu et al. [15] created a geographically temporally awareness hierarchical attention network (GT-HAN) to distinguish different user preferences.

Recent research proves that the Transformer outperforms other deep learning methods in trajectory prediction. Amirloo et al. [16] proposed LatentFormer, a transformer-based model able to predict future vehicle trajectories by leveraging a novel technique to model interactions among dynamic objects in the scene. Accounting for the interaction between vehicles, Yan et al. [17] proposed two spatial attention mechanisms to help the model understand the surrounding environment better and thus improve its prediction accuracy. Yu et al. [18] introduced the Spatio-Temporal grAph tRansformer (STAR) framework, a novel framework for spatio-temporal trajectory prediction based purely on a self-attention mechanism, with TGConv, a Transformer-based graph convolution mechanism. Dai et al. [19] proposed a novel neural architecture, Transformer-XL, which enables learning dependency beyond a fixed length without disrupting temporal coherence. Wang et al. [20] used a low-rank approximation method to approximate a self-attention mechanism, which maintains high performance while reducing the computational cost. Kitaev et al. [21] introduced reversible residual layers that reduce the memory consumption of the model and give the model the ability to handle larger datasets. Kong et al. [22] proposed the Spatial-Temporal Graph Attention Network (STGAT) for traffic flow forecasting. They demonstrated that STGAT can be generalized directly not only to graphs with an arbitrary structure, but also to completely unseen graphs. None of the existing deep-learning-based prediction methods consider the features in VEC. These deep learning methods need a large amount of storage and computational resources. However, edge servers have limited resources, which leads to the fact that these methods cannot be directly applied to VEC.

References

1. Liu, L.; Chen, C.; Pei, Q.; Maharjan, S.; Zhang, Y. Vehicular edge computing and networking: A survey. Mob. Netw. Appl. 2021, 26, 1145–1168.
2. Luo, Q.; Hu, S.; Li, C.; Li, G.; Shi, W. Resource scheduling in edge computing: A survey. IEEE Commun. Surv. Tutor. 2021, 23, 2131–2165.
3. Zhan, W.; Luo, C.; Min, G.; Wang, C.; Zhu, Q.; Duan, H. Mobility-aware multi-user offloading optimization for mobile edge computing. IEEE Trans. Veh. Technol. 2020, 69, 3341–3356.
4. Zhang, K.; Mao, Y.; Leng, S.; He, Y.; Zhang, Y. Mobile-edge computing for vehicular networks: A promising network paradigm with predictive off-loading. IEEE Veh. Technol. Mag. 2017, 12, 36–44.
5. Saeik, F.; Avgeris, M.; Spatharakis, D.; Santi, N.; Dechouniotis, D.; Violos, J.; Leivadeas, A.; Athanasopoulos, N.; Mitton, N.; Papavassiliou, S. Task offloading in Edge and Cloud Computing: A survey on mathematical. Comput. Netw. 2021, 195, 108177.
6. Yang, S.R.; Su, Y.J.; Chang, Y.Y.; Hung, H.N. Short-term traffic prediction for edge computing-enhanced autonomous and connected cars. IEEE Trans. Veh. Technol. 2019, 68, 3140–3153.
7. Alahi, A.; Goel, K.; Ramanathan, V.; Robicquet, A.; Fei-Fei, L.; Savarese, S. Social lstm: Human trajectory prediction in crowded spaces. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 961–971.
8. Han, P.; Wang, W.; Shi, Q.; Yang, J. Real-time short-term trajectory prediction based on GRU neural network. In Proceedings of the 38th Digital Avionics Systems Conference (DASC), San Diego, CA, USA, 8–12 September 2019; pp. 1–8.
9. Huang, M.; Zhu, M.; Xiao, Y.; Liu, Y. Bayonet-corpus: A trajectory prediction method based on bayonet context and bidirectional GRU. Digit. Commun. Netw. 2021, 7, 72–81.
10. Amichi, L.; Viana, A.C.; Crovella, M.; Loureiro, A.A. From movement purpose to perceptive spatial mobility prediction. In Proceedings of the 29th International Conference on Advances in Geographic Information Systems, Beijing, China, 2–5 November 2021; pp. 500–511.
11. Monreale, A.; Pinelli, F.; Trasarti, R.; Giannotti, F. Wherenext: A location predictor on trajectory pattern mining. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 28 June–1 July 2009; pp. 637–646.
12. Zhenjiang, D.; Jia, D.; Xiaohui, J.; Yongli, W. RTMatch: Real-time location prediction based on trajectory pattern matching. In Database Systems for Advanced Applications, Proceedings of the DASFAA 2017 International Workshops: BDMS, BDQM, SeCoP, and DMMOOC, Suzhou, China, 27–30 March 2017; Springer: Cham, Switzerland, 2017; pp. 103–117.
13. Zeng, J.; He, X.; Tang, H.; Wen, J. A next location predicting approach based on a recurrent neural network and self-attention. In Collaborative Computing: Networking, Applications and

Worksharing, Proceedings of the 15th EAI International Conference, London, UK, 19–22 August 2019; Springer: Cham, Switzerland, 2019; pp. 309–322.

14. Feng, J.; Li, Y.; Zhang, C.; Sun, F.; Meng, F.; Guo, A.; Jin, D. Deepmove: Predicting human mobility with attentional recurrent networks. In Proceedings of the 2018 World Wide Web Conference, Lyon, France, 23–27 April 2018; pp. 1459–1468.
15. Liu, T.; Liao, J.; Wu, Z.; Wang, Y.; Wang, J. A geographical-temporal awareness hierarchical attention network for next point-of-interest recommendation. In Proceedings of the 2019 on International Conference on Multimedia Retrieval, Ottawa, ON, Canada, 10–13 June 2019; pp. 7–15.
16. Amirloo, E.; Rasouli, A.; Lakner, P.; Rohani, M.; Luo, J. Latentformer: Multi-agent transformer-based interaction modeling and trajectory prediction. arXiv 2022, arXiv:2203.01880.
17. Yan, J.; Peng, Z.; Yin, H.; Wang, J.; Wang, X.; Shen, Y.; Stechele, W.; Cremers, D. Trajectory prediction for intelligent vehicles using spatial-attention mechanism. IET Intell. Transp. Syst. 2020, 14, 1855–1863.
18. Yu, C.; Ma, X.; Ren, J.; Zhao, H.; Yi, S. Spatio-temporal graph transformer networks for pedestrian trajectory prediction. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; pp. 507–523.
19. Dai, Z.; Yang, Z.; Yang, Y.; Carbonell, J.; Le, Q.V.; Salakhutdinov, R. Transformer-xl: Attentive language models beyond a fixed-length context. In Proceedings of the 57th Annual Meeting of the Association-for-Computational-Linguistics (ACL), Florence, Italy, 28 July–2 August 2019; pp. 2978–2988.
20. Wang, S.; Li, B.Z.; Khabsa, M.; Fang, H.; Ma, H. Linformer: Self-attention with linear complexity. arXiv 2020, arXiv:2006.04768.
21. Kitaev, N.; Kaiser, Ł.; Levskaya, A. Reformer: The efficient transformer. arXiv 2020, arXiv:2001.04451.
22. Kong, X.; Xing, W.; Wei, X.; Bao, P.; Zhang, J.; Lu, W. STGAT: Spatial-temporal graph attention networks for traffic flow forecasting. IEEE Access 2020, 8, 134363–134372.