Enabling Efficient Emulation of Internet-of-Vehicles on a Single Machine: Practices and Lessons

Xiaoxing Ming, Yicun Duan, Junyu Liu, Zhuoran Bi, Haoxuan Sun, Zilin Song, Xiangjun Peng, and Wangkai Jin(✉)

User-Centric Computing Group, Hangzhou, China
wangkaijin00@gmail.com

Abstract. The rise of Intelligent Vehicles and next-generation networks accelerates the advent of Internet-of-Vehicles (IoV), where each vehicle acts as a node and interconnects with other nodes for data sharing and processing. The primary challenges for evaluating Human-Vehicle Interactions (HVI) designs in the context of IoV are (1) computation on limited on-vehicle hardware resources; and (2) varied network connectivity in different settings. Moreover, existing emulators are highly customized for specific scenarios, which can limit the comprehensiveness of evaluations for IoV. To this end, we present a relatively general and portable emulation platform, that is designed for evaluating a variety of novel HVI designs under different IoV settings. Our emulation platform consists of two key components: 1) an automatic extractor to extract workload patterns in different IoV topologies and 2) configurable network settings to examine different HVI designs in various network conditions. We then characterize two IoV applications to examine the feasibility of our proposed emulation platform.

Keywords: Internet-of-Vehicles · Software Emulation

1 Background & Motivation

The rise of Intelligent Vehicles and next-generation networks accelerates the advent of Internet-of-Vehicles (IoV), where each vehicle acts as a node and interconnects with other nodes for data sharing and processing. Given its distributed nature and high costs to prototype, developers and researchers are prone to evaluate novel Human-Vehicle Interactions (HVI) designs for IoV in emulators.

However, current platforms and software impose limitations on exploring and analyzing IoV systems. To address this issue, several emulation platforms have been proposed for assessing the feasibility of advanced designs in IoV settings. For instance, OMNet++ [14] presents diverse protocols and wired/wireless network simulation models that are useful for discrete event systems. Simulation of Urban Mobility (SUMO) [8] concentrates on traffic simulations to examine traffic management strategies, while Vehicles in Network Simulation (VEINS) [10]

C. Stephanidis et al. (Eds.): HCII 2023, CCIS 1836, pp. 65–72, 2023.
https://doi.org/10.1007/978-3-031-36004-6_10

integrates both the network simulator OMNet++ and the road traffic simulator to evaluate the performance of Inter-Vehicle Communication (IVC) protocols. However, all these approaches focus on a particular application scenario and are highly customized for specific contexts. Therefore, no current platforms offer (A) the flexibility to adapt various application designs to the IoV context, and (B) the flexibility to explore a broad range of scenarios, such as geo-distributed scenarios.

To this end, we propose a relatively general and portable emulation platform that enables the development and evaluation of novel HVI designs in the context of IoV. Specifically, our platform has two main components that support HVI application designs in various scenarios, which are: 1) an automatic extractor to extract the key patterns of HVI workloads deployment on nodes within the IoV and 2) configurable network connections that enable the evaluation of HVI design in varied network conditions. Based on our emulation platform, we then characterize its impacts on two HVI applications in the context of IoV, which consist of a Deep-Neural-Network-based application and its Differentially-private version. We discuss our key takeaways throughout these two characterizations and discuss future works.

2 Design Overview

In this section, we first introduce our platform design in Sect. 2.1. Then, we brief the prototype implementation in Sect. 2.2.

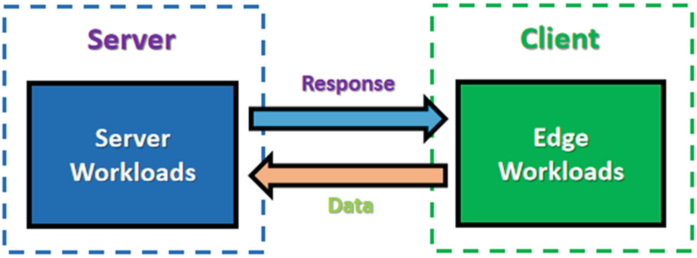

Fig. 1. Overview of workload distribution in the context of Internet-of-Vehicles

2.1 Study Design

Our platform design can be divided into two parts, which are automatic extractor and configurable network connection supports. First, it includes an automatic extractor that identifies and retrieves key patterns from real-world workloads. Second, the platform offers a configurable networking connection that supports the flexible enabling of different studies. Here is a brief explanation of these two components:

The automatic extractor is designed to identify and retrieve workload patterns. It starts to perform a static full-program analysis by identifying key components of data movements, specifically networking transfer/synchronization. The extractor only considers networking movements as networking connections have significantly higher delays than local data transfer. The extractor then organizes all significant data movements into a data sheet, recording all major behaviors and separating them into the server and edge workloads, with a particular focus on networking behaviors.

The configurable networking connection is supported by the platform in two ways. Firstly, the platform receives the generated data sheet from the automatic extractor and models the workloads in a server-client manner. Figure 1 illustrates the organization of the modeled workloads. Secondly, the platform offers a wide range of configurable parameters to support different study purposes. For instance, it can be configured with different numbers of connections (to examine the effects of concurrent services) and different response latencies (to examine the effects of geo-distributed scenarios).

2.2 Prototype Implementation

We build a prototype system using 2,000+ lines of C code, which we rigorously tested on a Linux system. Our prototype supports highly-concurrent emulation through the socket and connection pooling. To make the emulations more realistic and reduce the emulation latency, we leverage multiple parallelism-driven techniques, such as multi-threading, to increase throughput. Additionally, we provide customized support for porting high-level applications to the emulation, such as MySQL and Epoll.

In our emulation platform, we assumed that the server is always ready to be connected by edge vehicles for any type of request or query. It's worth noting that our emulation platform supports the interaction of multiple types of data, ranging from plain text to complex images. We believe such a feature can best reflect real-world scenarios where the communication between servers and clients is based on various data types.

3 Evaluations

We experimentally evaluate two representative IoV applications (i.e., DNN model inference and DNN model training with Differential-Private protection) on our platform. For experimental details, we discuss the workload settings and testing environment in Sect. 3.1, and then showcase our emulation results and takeaways for these two applications in Sect. 3.2.

3.1 Experimental Setup

Workload 1: Face2Multi-Modal. We consider positioning the pipeline of Face2Multi-modal in either (1) on-edge devices or (2) servers during services.

Therefore, there are two configuration settings in our experiments: 1) models are deployed on edge servers, which considers the scenarios that intelligent vehicles process the data in a local region and then synchronize with the corresponding server; 2) models are hosted on the servers, which considers a centralized server to process all the tasks in the server side and send results back to waiting vehicles. For each setting, it takes 20,000 images of 224×224 pixels as input and takes the heart rate and vehicle speed predicted by the model as output. And we also conduct 2000,5000, and 10000 times of concurrent connections for these two settings, to examine the impacts of the above strategies in the IoV context.

Workload 2: Differentially-Private Face2Multi-Modal Training. We also test the privacy-protected DNN training process whose training samples are obtained from the (1) centralized server or the (2) edge vehicles using Differentially-Private (DP) protected queries. For the first case, we conduct Global DP (GDP) queries on a centralized server to get the training data that is protected by the server (i.e., adding random noises). For the second case, the Local DP (LDP) method applies privacy protections locally before we train the per-vehicle DNN model. As for detailed GDP and LDP settings during training, we vary the ϵ value within the range [0.01, 1.2] to comprehensively characterize GDP and LDP for different privacy-protection levels. Face2Multi-modal uses BROOK dataset [4,6,9] for training under Differential Privacy settings.

Hardware Support: Our emulation platform of IoV is deployed on a single laptop, as the end goal of this work is to support the development and evaluation of novel HVI design in the context of IoV using accessible hardware resources. The detailed configurations of our selected laptop are listed as follows: The CPU is Intel(R) Core(TM) i7-8750H and the GPU is NVIDIA GeForce 1060. Ideal, the emulation platform has minimal requirements on the device and can be deployed on any machine.

3.2 Characterization Results

We present the characterization results of afore-mentioned two workloads respectively, and briefly summarize four important takeaways from the results.

Face2Multi-Modal on Edge or Server. Figure 2 reports our emulation results for testing two deployment patterns of Face2Multi-Modal on IoV. From the figure, we draw two key takeaways, which are presented as follows.

Key Takeaway 1: Latency-critical workload should be hosted on edge. As shown in Fig. 2, when the Face2Multi-Modal is deployed on the edge side, the average time consumption achieves the lowest level no matter how many clients are served concurrently. At concurrent numbers 2,000/5,000/10,000, the service time cost for the model deployed on edge is merely about 50%/63%/67% of that for the model on the server. This is because, from the perspective of

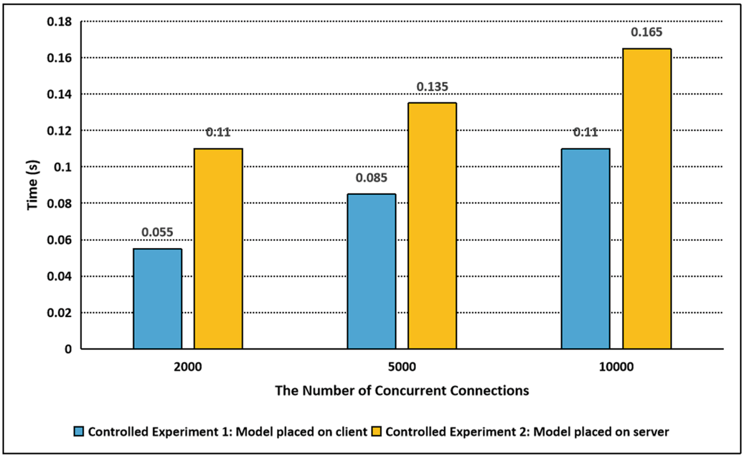

Fig. 2. Characterization results of Face2Multimodal in different client/server deployment settings on our emulation platform

time consumption, transferring data frequently between edge and server is less efficient than computing the DNN inference directly on edge. Therefore, in the context of IoV, it's efficient for us to deploy workload on edge when the latency is the users' primary concern.

Key Takeaway 2: Server should take the throughput-critical workload. Figure 2 demonstrates that the latency of hosting services on the server is less amenable to the increase in the number of concurrent connections. When the number of concurrent connections grows from 2,000 to 5,000, the latency of server-side services increases by approximately 22%, compared to that for services on edge (about 55% increment). Similarly, when the concurrent number reaches 10,000, the time cost for server-side workload suffers a 22% increase (compared to the latency with 5000 connections), while edge devices take about 30% more time to handle the workload. Hence, we should offload the throughput-critical workload to a server in the context of IoV.

DP-Protected Face2Multi-Modal Training. Figure 3 reports the emulation results of Face2Multi-Modal training in different Differential Privacy settings (i.e., GDP and LDP). From the figure, we draw two key takeaways.

Key Takeaway 3: In comparison to GDP, LDP shows its overwhelming advantages in the context of IoV. From Fig. 3, we can see a 1.7× validation accuracy gain when comparing the result of LDP and GDP. This may be because locally trained models fit better to the driver's features, which leads to higher accuracy. In light of this, we should employ LDP rather than GDP for privacy protection on IoV.

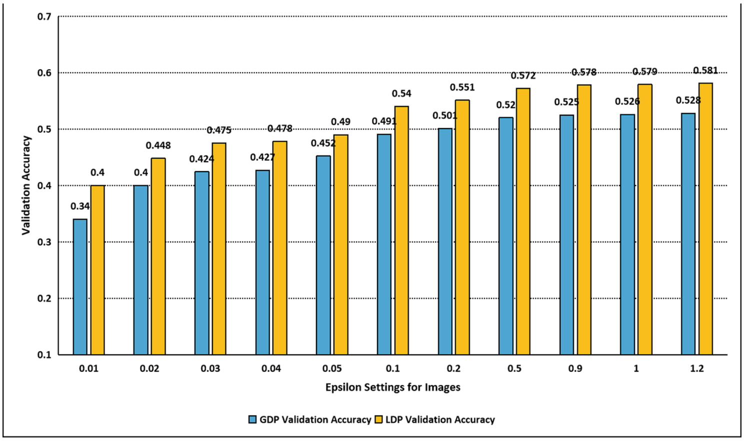

Fig. 3. Charaterization results of Differential-Privacy-enabled Face2Multimodal in different Differential Privacy protection settings on our emulation platform

Key Takeaway 4: An appropriate ϵ value of DP lies in the range [0.03, 0.5], which is expected to bring a reasonable tradeoff between privacy protection and inference accuracy. It's shown in Fig. 3 that the choice of ϵ value from an extremely low margin (i.e., [0.01, 0.03)) will greatly degrade the performance of Face2Multi-Modal. Improving the protection extent (ϵ) slightly from 0.03 to 0.01 takes the large cost of about 8% accuracy degradation. In the other margin (i.e., (0.5, 1.2]), the variance of ϵ brings about an inapparent effect on model accuracy, which means that choosing a ϵ value larger than 0.5 will not significantly benefit the accuracy, but compromises the privacy protection. Hence, we empirically verify that a preferred ϵ value should be located between 0.03 and 0.5.

4 Discussion and Future Work

The characterization results showcase the capability of our emulation platform to examine novel HVI applications in the context of IoV, by observing application behaviors in different server/client settings and evaluating parameter scopes for better tuning. The current takeaways we obtained are potentially applicable to other DNN-based HVI applications (e.g., [1,17]). In the future, we aim to add more functionalities to support the evaluation and characterization of various HVI designs and provide fine-grained emulation controls and detailed performance analysis to increase the usability of our platform. Some example works that can be deployed on our emulation platform to understand its performance in different IOV settings include 1) time-series-based analysis (e.g., driver style classifications [18], facial expressions distributions [15]) and 2) security and privacy protection designs [2,3,7,13]. Also, our emulation platform can be co-worked or integrated with existing IoV-related development toolkits (e.g., [11,12,16]), to provide a fast and glueless transition from prototype development to design emulation.

5 Conclusions

In this work, we present our emulation platform that is designed for developing and evaluating novel Human-Vehicle Interaction design in the context of Internet-of-Vehicles. We present the design and implementation details of our emulation platform and conduct two characterization studies to evaluate the feasibility of our platform. We then extract several key takeaways from the characterization studies. We also discuss potentials for future work of our platform.

Acknowledgements. We thank the anonymous reviewers from HCI 2023 and all members from the User-Centric Computing Group for their valuable feedback and comments. An earlier version of this work was positioned at [5] as a work-in-progress.

References

1. Bi, Z., et al.: FIGCONs: exploiting FIne-Grained CONstructs of facial expressions for efficient and accurate estimation of in-vehicle drivers' statistics. In: International Conference on Human-Computer Interaction (2023)
2. Duan, Y., et al.: Characterizing and optimizing differentially-private techniques for high-utility, privacy-preserving internet-of-vehicles. In: International Conference on Human-Computer Interaction (2023)
3. Duan, Y., et al.: Characterizing differentially-private techniques in the era of internet-of-vehicles. In: arXiv (2022)
4. Jin, W., et al.: BROOK Dataset: A Playground for Exploiting Data- Driven Techniques in Human-Vehicle Interactive Designs. Technical report- Feb-01 at User-Centric Computing Group, University of Nottingham Ningbo, China (2022)
5. Jin, W., et al.: Towards emulating internet-of-vehicles on a single machine. In: Automotive UI 2021: 13th International Conference on Automotive User Interfaces and Interactive Vehicular Applications, Leeds, United Kingdom, September 9–14, 2021 - Adjunct Proceedings, pp. 112–114. ACM (2021). https://doi.org/10.1145/3473682.3480275
6. Liu, J., et al.: BROOK dataset: a playground for exploiting data- driven techniques in human-vehicle interactive designs. In: HCII (2023)
7. Liu, J., et al.: HUT: enabling high-utility, batched queries under differential privacy protection for internet-of-vehicles. In: arXiv (2022)
8. Lopez, P.A., et al.: Microscopic Traffic Simulation using SUMO. In: ITSC (2018)
9. Peng, X., Huang, Z., Xu, S.: Building BROOK: a multi- modal and facial video database for human-vehicle interaction research. In: CHI workshop (2020)
10. Sommer, C., German, R., Dressler, F.: Bidirectionally coupled network and road traffic simulation for improved IVC analysis. In: IEEE TMC (2011)
11. Song, Z., et al.: First attempt to build realistic driving scenes using video-to-video synthesis in OpenDS framework. In: AutomotiveUI (2019)
12. Song, Z., et al.: Omniverse-OpenDS: enabling agile developments for complex driving scenarios via reconfigurable abstractions. In: HCII (2022)

13. Xu, S., et al.: Exploring Personalised Autonomous Vehicles to Influence User Trust. In: Cogn. Comput. (2020)
14. Andras Varga. "OMNeT++". In: Modeling and tools for network simulation (2010)
15. Wang, J., et al.: The Importance Distribution of Drivers Facial Expressions Varies over Time! In: AutomotiveUI (2021)
16. Wang, S., et al.: Oneiros-OpenDS: an interactive and extensible toolkit for agile and automated developments of complicated driving scenes. In: HCII (2022)
17. Xiong, Z., et al.: Face2Statistics: user-friendly, low-cost and effective alternative to in-vehicle sensors/monitors for drivers. In: HCII (2022)
18. Zhang, Y., et al.: Demystifying interactions between driving behaviors and styles through self-clustering algorithms. In: HCII (2021)