 CMC, vol.59, no.2, pp.375-387, 2019

A Data Download Method from RSUs Using Fog Computing in Connected Vehicles

Dae-Young Kim[1] and Seokhoon Kim[2, *]

Abstract: Communication is important for providing intelligent services in connected vehicles. Vehicles must be able to communicate with different places and exchange information while driving. For service operation, connected vehicles frequently attempt to download large amounts of data. They can request data downloading to a road side unit (RSU), which provides infrastructure for connected vehicles. The RSU is a data bottleneck in a transportation system because data traffic is concentrated on the RSU. Therefore, it is not appropriate for a connected vehicle to always attempt a high speed download from the RSU. If the mobile network between a connected vehicle and an RSU has poor connection quality, the efficiency and speed of the data download from the RSU is decreased. This problem affects the quality of the user experience. Therefore, it is important for a connected vehicle to connect to an RSU with consideration of the network conditions in order to try to maximize download speed. The proposed method maximizes download speed from an RSU using a machine learning algorithm. To collect and learn from network data, fog computing is used. A fog server is integrated with the RSU to perform computing. If the algorithm recognizes that conditions are not good for mass data download, it will not attempt to download at high speed. Thus, the proposed method can improve the efficiency of high speed downloads. This conclusion was validated using extensive computer simulations.

Keywords: Connected car, fog computing, data download, DSRC, machine learning.

1 Introduction

Connected vehicles perform wireless communication using On-Board Units (OBUs), and carry out data transmission by forming mobile ad-hoc networks (MANETs) with other vehicles and road side units (RSUs). The connected vehicles use wireless V2X communication. The V2X suite includes Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), Vehicle-to-Nomadic device (V2N), and Vehicle-to-Pedestrian (V2P) algorithms [Shrestha, Bajracharya and Nam (2018); Filippi, Moerman, Daalderop et al. (2016); Behravesh and Butler (2016); Song (2017)]. This connectivity is crucially important for an intelligent vehicle. The applications are classified into two types: safety and non-safety. Safety applications produce warning messages to prevent vehicle

[1] School of Information Technology, Daegu Catholic University, Gyeongsan-si 38430, Republic of Korea.

[2] Department of Computer Software Engineering, Soonchunhyang University, Asan-si 31538, Republic of Korea.

[*] Corresponding Author: Seokhoon Kim. Email: seokhoon@sch.ac.kr.

accidents. Low latency and high reliability are required. Non-safety applications are infotainment [Song (2017); Liu, Naik and Park (2017); Chen, Jiang and Delgrossi (2009); Kim, Oh and Kang (2017)]. Infotainment includes the traffic management services of a transportation system, and entertainment services. These services use Internet access for data storage, video streaming, and file downloading. Such applications do not require low latency or high reliability. Various services in a connected vehicle depend on downloaded information from other nodes or the Internet Cloud. The RSUs form a gateway for access to the Cloud. When a connected vehicle connects to an RSU, drivers or passengers can acquire large amounts of data, such as multimedia information, from the Cloud. Fig. 1 depicts the connected vehicle system.

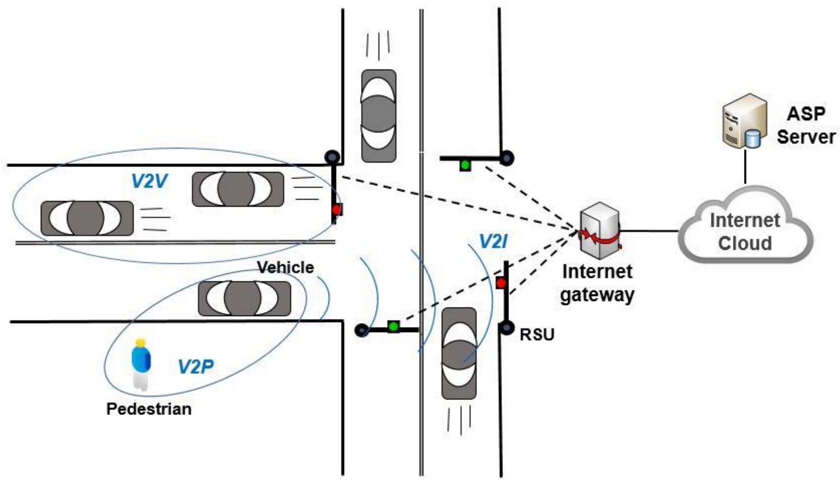

Figure 1: A connected vehicle system with V2X communication

For wireless access in connected vehicles, dedicated short-range communication (DSRC) and Long-Term Evolution (LTE) protocols are widely used. These services employ IEEE 802.11p and cellular protocols, respectively. DSRC and LTE can be integrated and used for heterogeneous vehicle networks. DSRC uses a wireless connection in the 5 GHz frequency band and supports a maximum 27 Mbps data rate. It has a maximum 1km transmission range [Song (2017); Liu, Naik and Park (2017); Chen, Jiang and Delgrossi (2009)]. DSRC has a control channel for short message transmission and service channels for IP packet transmission. In general, DSRC is suitable for MANET and is known to be valuable for short message transmissions for safety applications. LTE communication can be constantly connected to the Cloud, which allows infotainment applications to be served anytime and anywhere. Thus, LTE is the preferred protocol for infotainment applications [Song (2017); Xu, Li, Zhao et al. (2017)]. However, data transmission using LTE is costly. DSRC OBUs do not access the Cloud, but RSUs can access the Cloud via their infrastructure. When a connected vehicle enters an RSU zone, data can be downloaded at high speed and minimal cost using DSRC.

Although DSRC is cost-effective, it is a contention-based data transmission technique, as is CSMA/CA. It can be heavily influenced by network conditions [Song (2017); Liu, Naik and Park (2017); Fitah, Badri, Moughit et al. (2018)]. High-speed transmissions may fail even if connected to the RSU, depending on network conditions. Therefore, the method proposed here estimates the possibility of high-speed download from an RSU using DSRC, and tries to download at high speed if possible. To estimate the possibility of high-speed download, the algorithm employs machine learning and fog computing. Fog computing builds a mobile edge cloud using a fog server, and provides computing environments to edge devices via wireless. The result of learning in the fog server is applied to connected vehicles. This approach reduces the computing load of a connected vehicle. The role of fog computing in connected vehicles is to distribute computing loads from the Cloud to the mobile edge cloud. Data from a connected vehicle is stored in a fog server. The fog server processes the data and provides services from the Cloud, decreasing latency and increasing throughput [Shrestha, Bajracharya and Nam (2018)]. In addition, a fog server can collect information from vehicles around RSUs which can be used for learning and subsequent decision-making.

As previously discussed, connected vehicles can require very large amounts of data from the Cloud for their application services, particularly for infotainment applications. The proposed method performs decision-making to determine high speed downloading using network information from a vehicle when the vehicle encounters an RSU. This approach increases the success rate of high-speed downloading using DSRC in RSUs and reduces the usage of LTE in a high speed download. Thus, the proposed method can lead to cost-effective mass data transfer in connected vehicles.

The rest of this paper is organized as follows: Section 2 presents related work; in Section 3, the proposed method is described; in Section 4, performance evaluation using computer simulations is reported; finally, Section 5 concludes the paper.

2 Related work

Various technologies have been applied to connected vehicle applications. V2X communication is used to construct vehicle area networks (VANET) and provide connectivity to objects around a vehicle. For VANETs, Cloud computing is used. Vehicles generally obtain infotainment data such as transportation and multimedia information through Cloud services. Vast amounts of VANET data can be processed efficiently in the network. An edge cloud can be valuable for VANET data processing. Edge cloud computing moves the burden of data processing from the Internet Cloud to the edge cloud, and therefore provides services close to end users [Shrestha, Bajracharya and Nam (2018); Filippi, Moerman, Daalderop et al. (2016)].

2.1 V2X communication

V2X communication allows vehicles to exchange data with other vehicles or infrastructure. A cooperative intelligent transportation system has been built using V2X communication. The system efficiently controls traffic congestion, and effectively responds to traffic accidents. As mentioned earlier, both LTE and DSRC are based on IEEE 802.11p, and implement V2X communication. LTE is always connected to a base station via the cellular infrastructure, and provides access to the Cloud. The V2X communication of LTE relies on cellular infrastructure-based communication. The D2D protocol of the V2X study group in

3GPP is used for the V2X communication [Höyhtyä, Apilo and Lasanen (2018)]. To fully support V2X applications in LTE, several issues must be addressed, including handoffs between mobile network operators and cooperation among application service providers [Filippi, Moerman, Daalderop et al. (2016)].

DSRC is used for wireless access for vehicular environments (WAVE). It uses seven wireless channels in the 5.9 GHz frequency band, each of which has a 10 MHz bandwidth in the 75 MHz spectrum and 5 MHz in the guard band. According to the modulation scheme, it has a data rate of 6 to 27 Mbps. Its coverage has a maximum of 1km. DSRC has two types of wireless channels: a control channel (CCH) and a service channel (SCH). Short safety messages are exchanged during CCH intervals, and infotainment data is transmitted during SCH intervals. Each interval is 50 ms. V2X devices only operate CCH or switch between CCH and SCH [Behravesh and Butler (2016); Chen, Jiang and Delgrossi (2009)]. The WLAN protocol is modified for DSRC, which is an IEEE 802.11p standard. The IEEE 802.11p standard uses CSMA/CA for medium access control. To support priority in both CCH and SCH, it employs enhanced distributed channel access (EDCA) as specified in IEEE 802.11e [Song (2017); Fitah, Badri, Moughit et al. (2018)]. EDCA operates several queues according to priority, selects which frame to transmit through scheduling from the queues, and transmits the frame using the CSMA/CA protocol. In DSRC, there is a coexistence issue with WiFi (IEEE 802.11ac) in the 5.9 GHz frequency band, and this issue must be addressed in order to fully support V2X applications [Liu, Naik and Park (2017)].

There are various V2X applications in both the safety and the non-safety areas. Either LTE or DSRC can be applied depending upon the application. In general, LTE is preferred for infotainment applications, while DSRC is known to be superior to LTE for safety applications [Xu, Li, Zhao et al. (2017)]. LTE is preferred for infotainment because Internet access is always available. However, LTE is a costly communications technology. If LTE alone is used for high-volume data transmission, high levels of communication signals can be generated. Therefore, high-volume data transmission using DSRC is also used in the operation of RSUs. An RSU provides Internet access, and can be part of one of a variety of communication environments. For high-volume data transmission, it is necessary to confirm whether or not high speed data transmission in an RSU is possible. Fig. 2 shows LTE and DSRC communication for VANETs.

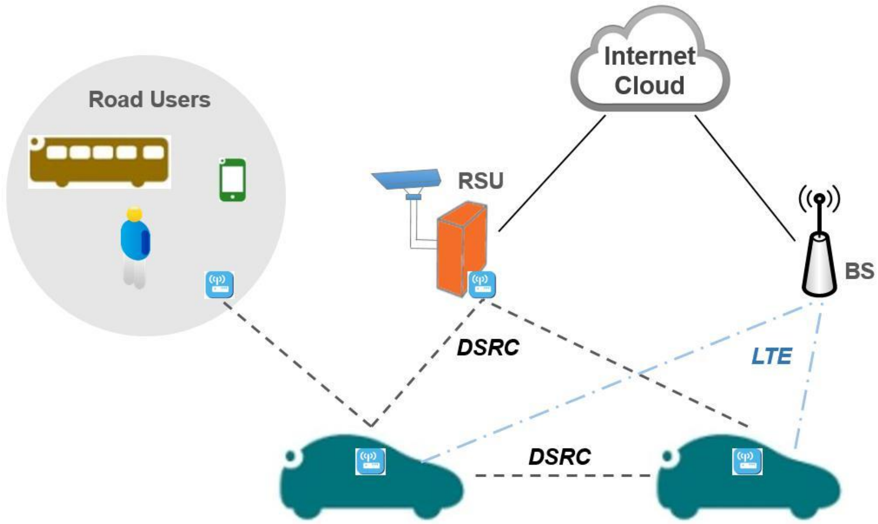

Figure 2: Connectivity of VANETs

2.2 Edge cloud computing

An edge cloud is formed near a wireless access network. It connects devices in the access network with the Internet Cloud, and distributes computing loads from the data core to the network edge. Because computing is performed at the network edge, service delays in the wireless network can be dramatically reduced. All data traffic is transmitted via the edge cloud, so the edge cloud can gather network status information. This information is used to ensure efficient system control [Sun and Ansari (2016); Mach and Becvar (2017); Ahmed and Ahmed (2016)].

To build an edge cloud in a vehicular network, fog computing [Shrestha, Bajracharya and Nam (2018); Roman, Lopez and Mambo (2018)] can be applied. By integrating a fog server with an RSU, an edge cloud can be constructed. The fog computing architecture consists of three tiers. Vehicle applications are in the first tier. First tier applications are for the wireless access network. In the second tier, the fog platform is placed. The fog platform interconnects the fog server with the RSUs. It provides data received from the Cloud to vehicles connected with RSUs. The third tier uses the Cloud as a data center, and when a vehicle requests data to download, it comes from the Cloud [Shrestha, Bajracharya and Nam (2018)]. Fig. 3 shows the fog computing architecture. In the edge cloud, the fog server supports resource virtualization. Applications can use the communication services regardless of their communication interfaces. Thus, different network connections can be provided for different vehicle applications. The computing capacity of a fog server is higher than that of the vehicles in the wireless access network. Vehicles can offload their computing loads to the fog server of the edge cloud. The fog server also provides storage for temporary data. Using this storage, it serves low latency and high bandwidth applications.

An RSU integrated with an edge cloud provides Internet access and collects vehicular network information. Using the network information, areas in which high speed data downloading is possible can be identified. The proposed method addresses techniques for downloading data in a high speed environment using DSRC in RSUs. The learning needed for decision making is carried out on the fog server in an edge cloud.

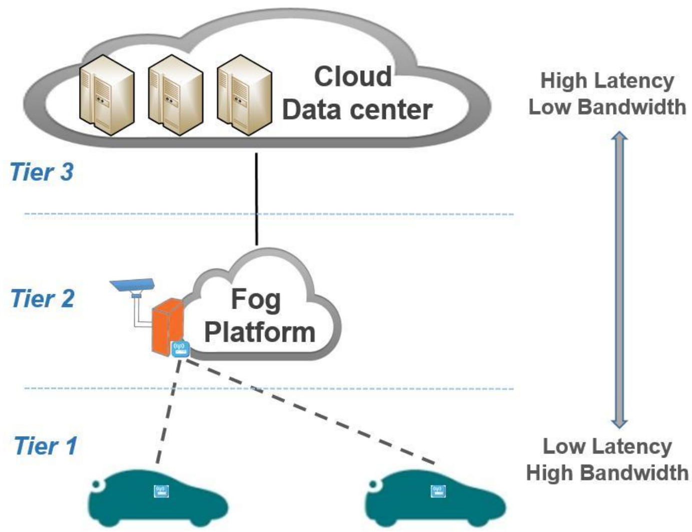

Figure 3: Fog computing architecture

3 Proposed data downloading approach

Connected vehicle applications for non-safety issues depend on downloading services from the Cloud. To maintain service quality for the applications, data downloading should be efficient. As discussed above, cellular-based connection can generate high costs, and thus it is most efficient to attempt to download data when a costless connection is possible. That is, a vehicle should access its service from Cloud using costless communications such as DSRC with an RSU, where possible. The costless connectivity environment is not, however, as stable as a cellular network. Attempts to download data from an unstable connection with an RSU will reduce service efficiency. Therefore, intermittent connectivity and support of network intelligence become major issues for VANET services using the Internet access. The proposed method employs logistic regression to determine whether large data should be downloaded from an RSU using DSRC at a particular point in time.

3.1 Learning for decision-making

Machine learning is often used for decision making. There are two major types of algorithms: supervised and unsupervised. Supervised learning uses a training dataset in which the data is labeled by the researcher. Algorithms for supervised learning are often used for classification or decision making. Unsupervised learning uses datasets that are unlabeled. Unsupervised learning is often used for grouping or clustering [Alsheikh, Lin, Niyato et al. (2014); Goodfello, Bengio and Courville (2016); Kim, Ko and Kim (2017); Zhang, Xie, Zhang et al. (2018)]. In the proposed method, a logistic regression algorithm is used with a supervised learning algorithm to determine strategies for data downloading. The algorithm is widely used to classify data with binomial distributions. Logistic regression uses a sigmoid function which generates values between 0 and 1. Putting the input data (z) through a sigmoid function (h) results in a probability value, which can be used for classification. The sigmoid function is shown in Eq. (1). If the output is greater than 0.5, the result is classified as 1. Otherwise, the result is classified as 0. That is, as shown in Eq. (2), the classification state represents as $y \in \{0,1\}$.

$$h(z) = g(w^T x) = \frac{1}{1 - e^{-w^T x}} \tag{1}$$

$$classification = \begin{cases} 1 & if \;\; h(z) > 0.5 \\ 0 & otherwise \end{cases} \tag{2}$$

Consider input data z as a linear system, composed of attribute values (x) and their weights (w) as shown in Eq. (3). In the equation, x_0 is 1 and w_0 is a bias.

$$z = w^T x = w_0 x_0 + w_1 x_1 + \cdots + w_n x_n \tag{3}$$

In the proposed method, the logistic regression algorithm is used to download large data. The input data for the decisions is the received signal strength and message exchange time. The decisions generated by the algorithm can be considered to represented probabilities. A y value of 1 represents the probability that large data downloading is feasible. In contrast, when y is 0 large-scale data downloading is probably infeasible. The probability is calculated as in Eq. (4).

$$P(y \mid x; w) = h(x)^y (1 - h(x))^{1-y} \tag{4}$$

Given training data m, the probability can be represented as in Eq. (5). It becomes a likelihood function of the weights.

$$\begin{aligned} L(w) &= \prod_{i=1}^{m} P(y^{(i)} \mid x^{(i)}; w) \\ &= \prod_{i=1}^{m} h(x^{(i)})^{y^{(i)}} (1 - h(x^{(i)}))^{(1-y^{(i)})} \end{aligned} \tag{5}$$

The likelihood function can be changed to the log likelihood function to avoid arithmetic underflow as shown in Eq. (6).

$$\begin{aligned} l(w) &= \log L(w) \\ &= \sum_{i=1}^{m} \left(y^{(i)} \log h(x^{(i)}) + (1 - y^{(i)}) \log(1 - h(x^{(i)})) \right) \end{aligned} \tag{6}$$

In the log likelihood function, weights w can be used to maximize the function. Using a gradient ascent algorithm, the appropriate values of w can be obtained. That is, w is updated by the gradient of the log likelihood function and the learning rate (α) as represented in Eq. (7).

$$\begin{aligned} w &:= w + \alpha \nabla_w l(w) \\ &:= w + \alpha (y - h(x)) x \end{aligned} \tag{7}$$

The weight (w_j) of the j^{th} attribute in the input data is continuously updated for the training data using a stochastic gradient ascent rule, as shown in Eq. (8). By applying the updated weights to the output, it is possible to decide whether to download large data using the RSU.

$$\begin{aligned}&for\ i = 1\ to\ m\ \{\\ &\quad w_j := w_j + \alpha\left(y^{(i)} - h(x^{(i)})\right)x_j^{(i)}, \quad \text{(for every } j\text{)} \qquad (8)\\ &\}\end{aligned}$$

3.2 Proposed VANET system

In the proposed VANET system, a fog server is integrated with an RSU to build edge cloud computing facilities. In the edge cloud, the fog server performs data-based learning for an RSU. The RSU periodically broadcasts its information in the CCH of DSRC. A connected vehicle receiving a message attempts to connect to the RSU by sending a request message to the RSU using the SCH of DSRC. The RSU receives the request message and responds by sending a response message in the SCH. Through this message exchange, a vehicle can connect to the RSU. The response message includes information about the weights, *w*, used for decision making in the connected vehicle. The weights are periodically updated by the fog server in the RSU. That is, the vehicle performs decision-making using the learning results from the fog server. Fig. 4 represents the proposed system model.

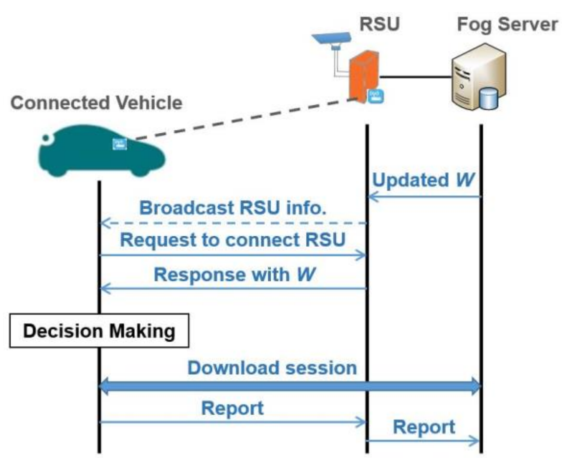

Figure 4: The proposed system model

The vehicle measures the time from when the request message is sent until the response message is received, and the received signal strength of the DSRC from the RSU. To determine the validity of large data downloading, it utilizes the function *h* described in Section 3.1 with the *w* from the received response message. The estimation value is obtained by applying the measured data, including the message exchange time and the received signal strength, to the algorithm. If the estimation value indicates that *y* is 1, the vehicle attempts to download the large dataset from the RSU. During the data download, the vehicle measures average throughput for 5 seconds and reports the measure to the fog server, along with the received signal strength and the message exchange time. This report continues during the download. After the download is finished, or upon disconnecting from the RSU, the fog server determines whether the state is success or not: y=1 or y=0. When the average throughput is greater than 1 MB/s, the label becomes success and y is 1. Otherwise, y is 0. In addition, the fog server gathers the vehicle's downloading information, such as the message exchange time and the received signal strength. This information and the label state are used to update the weights. Fig. 5 represents pseudocode of the proposed system.

```
<Vehicle>
1. If a vehicle enters the RSU zone
2.     sends Request to connect
3.     receives Response with the learning weights w
4.     applies w to the prediction function h
5.     If the prediction result is 1
6.         performs high speed downloading in DSRC
<Fog server>
7. If high speed downloading is finished
8.     update downloading information as a training data
9.     update w by learning with the updated training set
```

Figure 5: Pseudocode of the proposed system

Using the proposed method, a vehicle can choose an appropriate DSRC connection for mass data download. While waiting at the RSU zone for two or three minutes, the vehicle can receive large amounts of data using a costless connection. Thus, the proposed method can therefore improve the quality of services in DSRC. In addition, if a vehicle attempts to download large amounts of data when the prediction of the state for downloading is not good, vehicular applications can exploit HTTP Range Requests [Fielding, Lafon and Reschke (2014); Yun and Chung (2015); Kim, Kim, Han et al. (2016)]. HTTP Range Requests enable the vehicle to request data transmission in a specific byte range. Therefore, vehicular applications can download data using both LTE and DSRC, as shown in Fig. 2. This approach can reduce download time, but increases the usage rate of the cost connection. Thus, the proposed approach is important for a VANET system that depends on downloaded information.

4 Performance evaluation

Performance evaluation for the proposed method was carried out using computer simulations. The simulator was implemented with the SMPL library, an event-driven simulation library for C [MacDougall (1987)]. The simulation model functions as follows: The simulation time was set to 100,000 seconds. A vehicle attempts to download when it enters the RSU zone. The download generation of a vehicle is assumed to have a Poisson distribution, and to occur with an average of 1,800 seconds. The wireless channel of a DSRC is modeled by a two-state Markov chain [Trivedi (2002)]. The channel state may be good or bad, and state transition occurs according to a given probability. In the simulation, the transition probability (p) from good to bad was set to 0.3 and the transition probability (q) from bad to good was set to 0.4. Fig. 6 shows the channel state model. In the good state, the signal strength is randomly selected between -80 and -40 and throughput is randomly selected between 1 Mbps and 27 Mbps. The message exchange time is calculated using the throughput and processing times in the fog server. The processing time is assumed to be 2 ms. In the bad state, congestion is triggered. In the congestion state, the signal strength is randomly selected between -80 and -40. The throughput is randomly selected between 300 Kbps and 1 Mbps. In the bad state that is not congestion, the signal strength is randomly selected between -100 and -75. The throughput is randomly selected between 200 Kbps and 1 Mbps. Download data is set to 150 MB and a message for the request and response is set to 1,000 bytes.

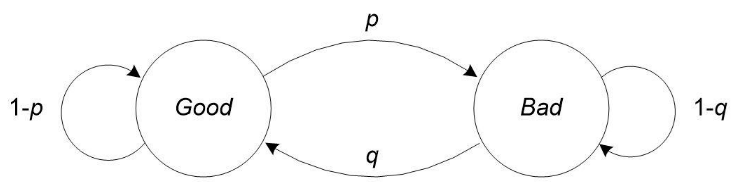

Figure 6: DSRC channel state model

In the simulation, the average download time and successful download rate are used as performance metrics. If a download is completed within 3 minutes, the download is considered successful. As shown in Fig. 4, when a vehicle enters the RSU zone, it measures the signal strength and the message exchange time and performs decision making for the mass data downloading. The system parameters (that is, the learning result: weights and bias) for the decision making are delivered from the fog server. The measured data is reported to the fog server and can be used to update training examples. For learning of the network prediction, a training set maintains 1,000 training data and has 0.01 as a learning rate. The learning algorithm is implemented with Python and TenserFlow library [TensorFlow (2018)]. The system parameters as the result are applied to the simulation. The accuracy of the logistic regression algorithm for the fog server is measured to 92%.

Fig. 7 depicts the average successful download rate of the DSRC. The proposed method shows an almost 90% success rate of large data downloads. In contrast, the conventional method which does not consider network conditions shows 40%~50%. Because the existing method does not consider DSRC state, a vehicle attempts to request data download when it enters the RSU zone. Thus, the existing method has a low success rate for large data downloads. The proposed method relies upon the learning algorithm for decision making. If the accuracy of the learning algorithm is improved, the success rate for large data download can be increased even further.

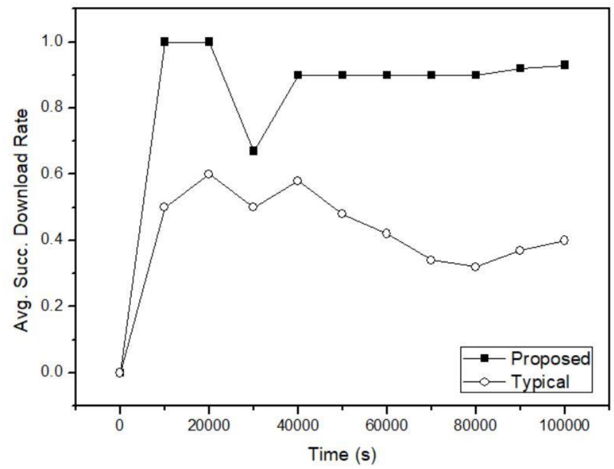

Figure 7: Average successful download rate

Fig. 8 shows the average download time when vehicular applications use the DSRC. The proposed method determines whether to request data download according to extant download conditions. It guarantees the quality of mass data downloading. Thus, the average data download time can be kept constant, as shown in the Fig. 8. The existing method attempts to download regardless of download conditions. That is, even under very low data rates of the DSRC, the existing method performs data downloading even when this action increases the average downloading time. Thus, our method shows overall higher data downloading times than the proposed method. In mass data download using a costless connection, the existing method often requires more than three minutes to download data. This means that the probability of successful downloading of large volumes of data in the RSU zone is low. Therefore, the role of the proposed method is particularly valuable for non-safety vehicular services that depends on the downloading of large amounts of data.

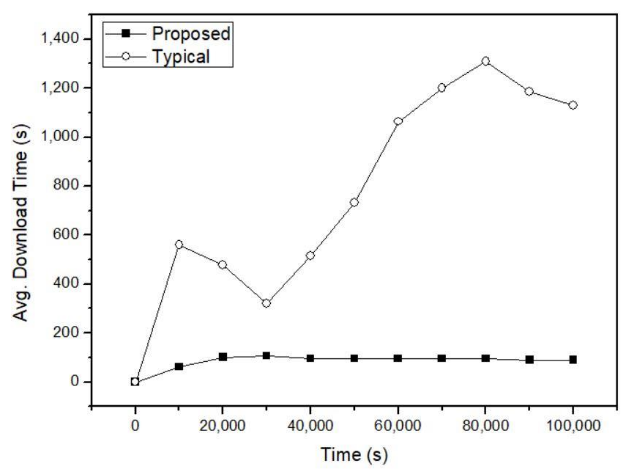

Figure 8: Average download time

5 Conclusion

Applications of connected vehicles depend on downloading data. Particularly, infotainment services of the applications require mass data download. The mass data downloading using costless connection such as DSRC should be completed while a vehicle stays in the RSU zone. Thus, checking download condition for mass data is important. If download condition is not considered before data download, service quality will be decreased by shortage of the received data for services. The proposed method attempts to download data considering download condition through results of a machine learning algorithm. The machine learning algorithm is performed in the fog server of edge cloud and uses vehicles' measured data for learning. Therefore, the proposed method improves success rate of the mass data downloading in the DSRC and reduces average download time for the mass data than the existing method.

Acknowledgement: This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2016R1D1A1B03931406), and this work was supported by the Soonchunhyang University Research Fund.

References

Ahmed, A.; Ahmed, E. (2016): A survey on mobile edge computing. *Proceedings of the IEEE International Conference on Intelligent Systems and Control.*

Alsheikh, M. A.; Lin, S.; Niyato, D.; Tan, H. P. (2014): Machine learning in wireless sensor networks: algorithms, strategies, and applications. *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 1996-2018.

Behravesh, E.; Butler, A. (2016): Evaluation of the IEEE 802.11p multi-channel operation in vehicular networks. *PeerJ Preprints*, vol. 4.

Chen, Q.; Jiang, D.; Delgrossi, L. (2009): IEEE 1609.4 DSRC multi-channel operations and its implications on vehicle safety communications. *Proceedings of the IEEE Vehicular Networking Conference*.

Fielding, R.; Lafon, Y.; Reschke, J. (2014): Hypertext transfer protocol (HTTP/1.1): range requests. *IETF RFC* 7233.

Filippi, A.; Moerman, K.; Daalderop, G.; Alexander, P. D.; Schober, F. et al. (2016): Ready to roll: why 802.11p beats LTE and 5G for V2X. *WhitePaper by NXP Semiconductors, Cohda Wireless and Siemens*.

Fitah, A.; Badri, A.; Moughit, M.; Sahel, A. (2018): Performance of DSRC and WIFI for intelligent transport systems in VANET. *Procedia Computer Science*, vol. 127, pp. 360-368.

Goodfello, I.; Bengio, Y.; Courville, A. (2016): *Deep Learning*. The MIT Press.

Höyhtyä, M.; Apilo, O.; Lasanen, M. (2018): Review of latest advances in 3GPP standardization: D2D communication in 5G systems and its energy consumption models. *Future Internet*, vol. 10, no. 1.

Kim, D. Y.; Ko, D.; Kim, S. (2017): Network access control for location-based mobile services in heterogeneous wireless networks. *Mobile Information Systems*, vol. 2017.

Kim, H. S.; Kim, I.; Han, K.; Kim, D.; Seo, J. S. et al. (2016): An adaptive buffering method for practical HTTP live streaming on smart OTT STBs. *KSII Transactions on Internet and Information Systems*, vol. 10, no. 3, pp. 1416-1428.

Kim, Y.; Oh, H.; Kang, S. (2017): Proof of concept of home IoT connected vehicles. *Sensors*, vol. 17, no. 6.

Liu, J.; Naik, G.; Park, J. M. (2017): Coexistence of DSRC and Wi-Fi: impact on the performance of vehicular safety applications. *Proceedings of the IEEE Conference on Communication*.

MacDougall, M. H. (1987): *Simulating Computer Systems, Techniques and Tool*. The MIT Press.

Mach, P.; Becvar, Z. (2017): Mobiled edge computing: a survey on architecture and computation offloading. *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1628-1656.

Roman, R.; Lopez, J.; Mambo, M. (2018): Mobile edge computing, Fog et al.: a survey and analysis of security threats and challenges. *Future Generation Computer System*, vol. 78, pp. 680-698.

Shrestha, R.; Bajracharya, R.; Nam, S. Y. (2018): Challenges of future VANET and cloud-based approaches. *Wireless Communications and Mobile Computing*, vol. 2018.

Song, C. (2017): Performance analysis of the IEEE 802.11p multichannel MAC protocol in vehicular ad hoc networks. *Sensors*, vol. 17, no. 12.

Sun, X.; Ansari, N. (2016): Edge IoT: mobile edge computing for the internet of things. *IEEE Communications Magazine*, vol. 54, no. 12, pp. 22-29.

TensorFlow (2018): Open source machine learning library.https://www.tensorflow.org/.

Trivedi, K. S. (2002): *Probability and Statistics with Reliability, Queuing and Computer Science Applications*. Wiley.

Xu, Z.; Li, X.; Zhao, X.; Zhang, M. H.; Wang, Z. (2017): DSRC versus 4G-LTE for connected vehicle applications: a study on field experiments of vehicular communication performance. *Journal of Advanced Transportation*, vol. 2017.

Yun, D.; Chung, K. (2015): Rate adaptation for HTTP video streaming to improve the QoE in multi-client environments. *KSII Transactions on Internet and Information Systems*, vol. 9, no. 11, pp. 4519-4533.

Zhang, J.; Xie, N.; Zhang, X.; Yue, K.; Li, W. et al. (2018): Machine learning based resource allocation of cloud computing in auction. *Computers, Materials & Continua*, vol. 56, no. 1, pp. 123-135.