QoE-Driven Big Data Architecture for Smart City

Xiaoming He, Kun Wang, Huawei Huang, and Bo Liu

Providing satisfactory QoE will become the major challenge in the big-data-based smart city. To enhance the QoE, the authors propose a novel big data architecture consisting of three planes: the data storage plane, the data processing plane, and the data application plane.

Abstract

In the era of big data, the applications/services of the smart city are expected to offer end users better QoE than in a conventional smart city. Nevertheless, various types of sensors will produce an increasing volume of big data along with the implementation of a smart city, where we face redundant and diverse data. Therefore, providing satisfactory QoE will become the major challenge in the big-data-based smart city. In this article, to enhance the QoE, we propose a novel big data architecture consisting of three planes: the data storage plane, the data processing plane, and the data application plane. The data storage plane stores a wide variety of data collected by sensors and originating from different data sources. Then the data processing plane filters, analyzes, and processes the ocean of data to make decisions autonomously for extracting high-quality information. Finally, the application plane initiates the execution of the events corresponding to the decisions delivered from the data processing plane. Under this architecture, we particularly use machine learning techniques, trying to acquire accurate data and deliver precise information to end users. Simulation results indicate that our proposals could achieve high QoE performance for the smart city.

Introduction

The smart city is an innovation of the physical city with a high integration of advanced monitoring, sensing, communication, and control technologies aiming to provide real-time, interactive, and intelligent urbanization services to end users. With the rapid implementation of smart cities, large amounts of various sensing equipments have been deployed everywhere. As a consequence, such sensing devices are producing ever growing data, leading to the size of big data evolving from gigabytes to terabytes, and then to petabytes and even exabytes in the future. Extracted from these big data in the smart city, the enormous amount of meaningful information can not only enhance the quality of smart city from the perspectives of urbanization and industrialization, but also provide high quality of experience (QoE) for big data applications/services, since end users expect higher quality of information than in the traditional physical city.

Recently, QoE has been defined by the International Telecommunication Union — Telecommunication Standardization Sector (ITU-T) as *the overall acceptability of an application or service* subjectively perceived by end users. This is because the utility and/or expectations of the applications/services are implemented based on end users' personalities and current situations. In summary, the general understanding of QoE is largely the same: QoE is a new measurement for smart city services determined by the precision of information.

QoE has been applied to various scenarios. For example, Su *et al*. [1] introduced a data-driven architecture to enhance personalized QoE for fifth generation (5G) networks and proposed a two-step QoE modeling method to capture the strength of the relationship between end users and services. Rahman *et al*. [2] proposed a rate adaptive algorithm to improve video quality and guarantee a promised QoE by observing the available throughput and managing the playback buffer. Tran *et al.* [3] proposed a novel QoE-driven energy-aware multi-path content delivery approach for mobile phones. Su *et al*. [4] introduced video broadcasting technologies to support a wide variety of multimedia devices interacting with video contents and reach heterogeneous QoE in smart city.

Inspired by the literature, we can see that it is necessary to use powerful approaches to analyze the big data and extract precise information to the satisfaction of QoE for applications/services of big data. But the challenge is that the processing of big data in smart cities brings stringent requirements in big data technologies due to the vast volume and variety. Compared to the traditional processing methods, machine learning techniques have some unique advantages in the extraction and delivery of big data services. Moreover, with advanced manipulations, deep learning and reinforcement learning techniques could achieve high data rate and precision. Many researchers have applied deep learning techniques in smart cities. As a typical example, Polishetty *et al.* [5] discussed a signature-based feature technique as a deep convolution neural network in the cloud platform for board location, segmentation, and character detection. Overall, machine learning techniques provide promising support for the processing of big data for smart cities, thereby providing satisfactory QoE for big data applications/services.

The QoE of applications/services is in every aspect of smart cities, helping business gain improved performance, enhancing health-care through improved preventive services, and benefiting transportation systems. We particularly focus

Digital Object Identifier:
10.1109/MCOM.2018.1700231

Xiaoming He is with Nanjing University of Posts and Telecommunications; Kun Wang (corresponding author) is with Nanjing University of Posts and Telecommunications and also with Hong Kong Polytechnic University; Huawei Huang is with Hong Kong Polytechnic University; Bo Liu is with La Trobe University.

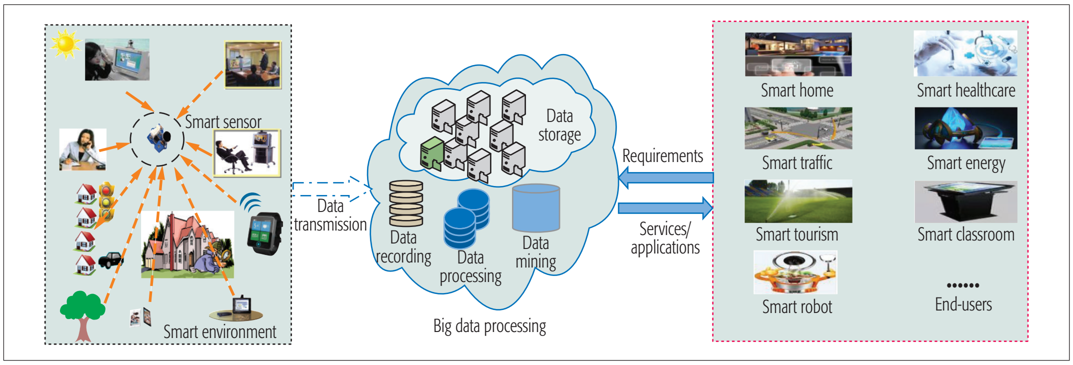

Figure 1. Big Data in the context of smart city.

on the issue of QoE satisfaction for smart cities in this article. With the goal of processing big data in smart cities, we propose a big data architecture that consists of three planes: the data storage plane, data processing plane, and data application plane. In short, the data storage plane stores all types of data collected from sensors. The data processing plane filters, analyzes, processes, and stores the types of data aiming to extract meaningful data to end users. The data application plane delivers the meaningful data to end users subject to their QoE requirements. Based on this architecture, one can extract the meaningful data for applications/services with satisfactory QoE. Finally, the major contributions of our work are summarized as follows:

- In the context of the smart city, emphasizing satisfactory QoE toward end users, we propose a novel big data architecture to process the large amount of raw data for extracting meaningful information.
- Based on the proposed architecture, we then devise a Spark and deep-learning-based greedy algorithm to improve the QoE level of the extracted information.
- Finally, we highlight several challenging open issues under our proposed big data architecture to offer potential future research directions.

The remainder of our article is organized as follows. The next section describes the challenges of big data and applications/services in smart cities as well as our motivations. The third section specifies the proposed hierarchical big data architecture for smart cities. A case study on content delivery is then presented. Next, we discuss the open issues. Finally, we conclude this article.

Big Data in Smart Cities

In this section, we mainly discuss the challenges of big data in the context of smart cities and our motivations.

Big Data Generation

A smart city needs to be based on the digital city and associated with the physical city using the ubiquitous network of sensors. A variety of sensing devices produce various data quickly, as shown on the left of Fig.1.

Furthermore, following the gradual construction and improvement of a smart city, mankind and all kinds of sensing equipment will generate a mounting number of data making the growing size of big data evolve from gigabytes to exabytes. At any time, people are exchanging data and information online. The type of generated data differs due to the deployed system. A great variety of generated data with meaningful information provides the foundation of applications/services feeding to end users.

Big Data Services with QoE

As shown in Fig. 1, the intelligent applications/services of big data offer the potential for the end users of a smart city to obtain meaningful information extracted from the large amounts of data collected through all kinds of sources. Different end users of a particular community can share the information with QoE by registering a big data architecture, which is specified in the next section. Then civilians and end users can acquire relevant information concerning tourism, and other related services. Typically, big data can bring key applications/services to the following areas.

Smart Grid: In the smart grid environment, a variety of data will be generated from a lot of data sources, such as the power utilization habits of customers, situational awareness of the phasor measurement data, and a wide range of intelligent meter measurement of energy consumption data and other fields [6]. Effective and timely utilization of big data from a smart grid can help administrators decide the level of power supply while meeting end users' demands adaptively.

Smart Healthcare: In the past two or three decades, an ocean of big data has been produced in the healthcare field [7]. Proper healthcare data can be used to cure diseases and predict epidemics, as well as to improve quality of life and avoid avertable death.

Smart Tourism: The applications/services of big data in intelligent tourism are the most direct embodiment of the change of management direction. For a tourism company, the adoption of big data will effectively change the passive situation of traditional tourism supervision, and realize modern science and technology management.

Challenges to Be Addressed and Our Motivations

Although the big data in smart cities brings end users applications/services with QoE, we are facing the challenges of retrieving the precise data of big data acquired from an ocean of data with diverse characteristics, and accurate information should be sent to end users with satisfactory QoE.

Currently, large amounts of data are being generated by different data sources. Given that the data generated from cities continuously grows, the underlying infrastructure has not provided sufficient capability of storing, processing, and analyzing big

At any time, people are exchanging data and information online. The type of generated data differs due to the deployed system. A great variety of generated data with meaningful information provides the foundation of applications/services feeding to end users.

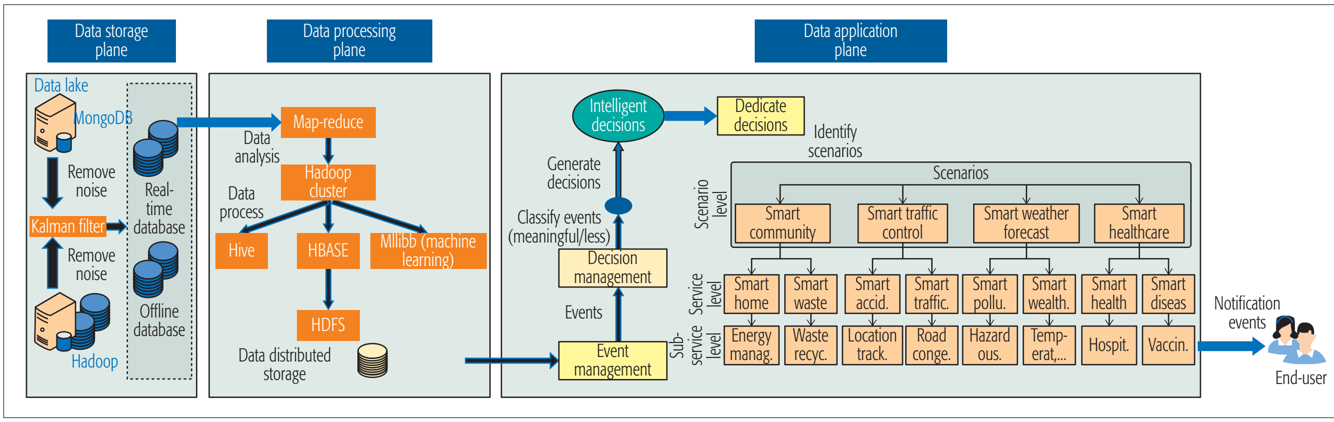

Figure 2. The proposed big data architecture for smart cities.

data. Furthermore, although effective analysis can dig precise information from the ocean of data, and the utilization of big data plays a critical role for the applications/services with satisfactory QoE, unfortunately, we find that an integrated big data architecture that can handle the entire source-destination business flow is still missing by conducting a comprehensive review of the state-of-the-art technologies. To fill this gap, this article aims to propose such an integrated big data architecture. The following challenging characteristics must be considered to realize it. First, the architecture needs to guarantee the efficient storage of various structured, unstructured, and semi-structured data. Second, the architecture should process both the real-time data and the historical data, and ensure that the aggregated results are updated in an incremental and scalable manner as data are generated by smart cities continuously. Also, the architecture should be flexible to extend its capacity of data processing. Third, the architecture should allow sharing the processed and analytic results of big data across applications/services within the specified community.

In the next section, we discuss how we deal with these issues with a proposed big data architecture for smart cities.

Big Data Architecture for Smart Cities

In this section, we present the featured big data architecture shown in Fig. 2. In particular, this architecture includes a data storage plane, a data processing plane, and a data application plane, each of which is specified as follows.

Data Storage Plane

A *data lake* provides huge data storage capability for the *data storage plane*. As defined by Wikipedia, a data lake is *a storage repository that contains a vast amount of raw data in the native format until it is required*. Once a data access request arises, the data lake will be queried for the relevant data, which is then analyzed to yield an answer.

Since the raw data is generated at ever increasing speed and the overall data volume is increasing, these data must be collected as fast as possible and stored in a cheap manner. The best situation is that none of the data is lost. Normally, the reality is that the value density of such raw data is very low, and the true value of these data is always unknown when it is first captured. Therefore, an infrastructure that can collect and store it at a low cost is desired. With the adoption of both MongoDB (Not Only Sql DataBase) and Haddop distributed file system (HDFS), the database of our architecture can keep low storage cost and achieve fast response. Particularly, data will be stored in a raw form, until it is queried. However, raw data need be transformed to make them useful. In our design the enormous amount of raw data stored in a data lake is filtered by fusion mechanisms to obtain valuable real-time and offline data. A Kalman filter is used to perform data filtration [8], which removes noise from the raw data. In the next procedure, referred to in [9], the data in both the real-time database and offline database are delivered to a Hadoop framework for further processing.

Data Processing Plane

The data processing plane acts as a mediator between the data storage plane and the application plane. Since critical processes such as data mapreduce, analyzing, processing, and storing events take place in this plane, it can be viewed as the brain of the proposed architecture.

The MapReduce under Hadoop framework works in two steps. First is the mapping process, where the set of filtered data is converted into another format. The next step is the reduce process, which combines the data created in the mapping process and results in a set of values that are reduced in amount. Note that data storage and processing are the major operations in this plane. As shown in Fig. 2, this plane is designed to utilize multiple techniques to facilitate the above requirements. In detail, the storage demand of this plane is facilitated by HDFS, which is the primary storage component of Hadoop. Since the storage of HDFS is distributed, it augments MapReduce execution on smaller subsets of big data clusters. In addition, HDFS enables the scalability of big data processing demands. After the data are imported into Hadoop clusters, we can use Mllib to conduct the analysis and mining of big data.

Then, to enable the autonomous decision making mechanism, the real-time read/write functionality over the complete cluster is crucial. Hence, HBASE is used to enhance the processing speed as it offers real-time lookups, in-memory caching, and server side programming. Further, it also supports fault tolerance. Hive provides querying and managing functionality over the large amount of data that resides on the Hadoop cluster. Finally, the derived intelligent decisions are transferred to the data application plane.

Data Application Plane

Likewise, the application plane is the mediator between the data processing plane and the end users. The application plane is divided into three fine-grained levels: the scenario level, services level, and sub-services level. The scenario level is the boundary at the data processing plane, and the sub-services level acts as the boundary for end users. The autonomous decisions from the data processing plane are dedicated to the specific scenario services.

The intelligent decisions of the data processing plane describe the decision according to a shared vocabulary (ontology). The ontology is used to dedicate the events throughout the application plane. The respective scenarios distinguish the meaningful events and the meaningless events. The meaningful events are stored at the scenarios level and are forwarded dedicated to the recipients, whereas the meaningless events are not moved further. Sequentially, the corresponding service event's component receives the dedicated event from the scenario events. For example, the service events smart home and waste management are readily available to receive the scenario events from the smart community development scenarios. Similarly, the service events are further categorized into sub-service events. The sub-service event level generates the respective event and transmits it to the embedded notification component. Finally, the notification component determines the specific recipient with respect to the generated event. Accordingly, it notifies the end user with the generated event for the event execution.

Case Study: Content Delivery in a Smart City

In this section, we present a content delivery scenario as a case study of applying the proposed big data architecture in a smart city. In such a scenario, shown in Fig. 3, system components include content servers, content processing units, and end-user devices. Content servers feed the raw big data to the data processing units, which return the precise contents. Then end users can deliver their requirements and feedbacks to the content servers through the middle data application plane. Consequently, the data processing plane keeps adapting the content presentation taking the QoE into account toward end users under different services. In this manner, our proposed big data architecture yields meaningful contents, and distributes proprietary contents to end users with QoE.

The Measurement of QoE Level

For our case study, the QoE is viewed as a crucial metric for applications and services in the smart city. With regard to end-user preference and quality of service requirements, we then specify how the QoE level should be evaluated in the scenario of a content delivery application for the case study. Note that we are not creating a new QoE model in this article. Our goal is to study the various QoE performance of different content-extracting algorithms under our proposed big data architecture. To this end, we adopt the following QoE measurement model [10], which returns a QoE value if the precise content (denoted by P) is given: $QoE_f(P) = a \cdot e^{b \cdot P_f} + r$, where a = 29.001, b = 2.863, r = 1.235 are the coefficients of the model, while P_f is the precise contents of the end user with the fth subscription. The reason we adopt this model is that it considers both application and algorithm parameter for evaluating the QoE, offering us an appropriate measurement approach.

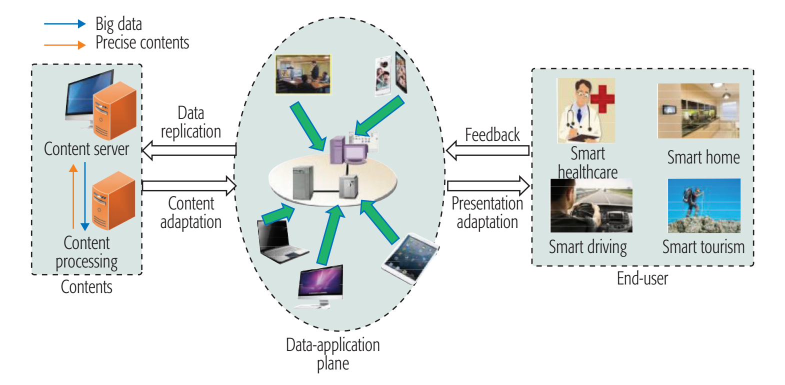

Figure 3. The use case of content delivery in a smart city.

Spark and Deep Learning Based Greedy Algorithm

Now the problem is to design an effective algorithm for extracting the precise contents with the satisfactory of QoE for smart cities. Deep learning [11] is a well-known dependable tool for big data analysis. Specifically, it can offer highly precise results of big data analysis. Spark [11] is an open source platform for scalable computing on datasets. It aims to speed up big data decision making by the learning of deep models parallel to high-performance computing sets. Therefore, we choose a Spark- and deep-learning-based approach to mine the meaningful contents quickly, aiming to meet the QoE requirements of end users. In more detail, simulated neurons and synapses consist of a deep learning model that can be used to train and learn a variety of features from available samples of big data. The available deep learning model can process unknown streaming samples of big data. Because a deep model normally includes many hidden layers and a large amount of parameters that are difficult to train quickly, a greedy algorithm of layer-by-layer learning [12] has been designed with the illustration in Fig. 4. The critical procedures of this greedy algorithm are explained as follows.

Generative Pre-Training: This stage only requires unlabeled data that is often massive and cheaply collected from the smart city by crowdsourcing. Figure 4 illustrates the sensible tuning of a deep learning model. First, a layer of initial neurons are trained using the unlabeled samples of data. Each of the following layers includes both encoding and decoding functions, which are to learn the input data structure. The encoding function leverages the input data and other layer parameters to yield a set of new features. In contrast, the decoding function mainly utilizes these features to reconstruct the input data. As a result, the first set of features is generated at the layer output. Then the second layer of neurons are built on the roof of the first layer, where the output of the first layer is fed as the input of the second layer. This procedure is repeated by building more layers until a desired deep model is formed. Accordingly, tremendous complex features can be learned in each layer based on the features generated at the previous layer.

In smart cities, although the application/service content can be delivered to end users, the quality of applications/services is determined by the precise data. Therefore, we selected machine learning algorithms to process the big data for our proposed architecture so as to extract the precise data.

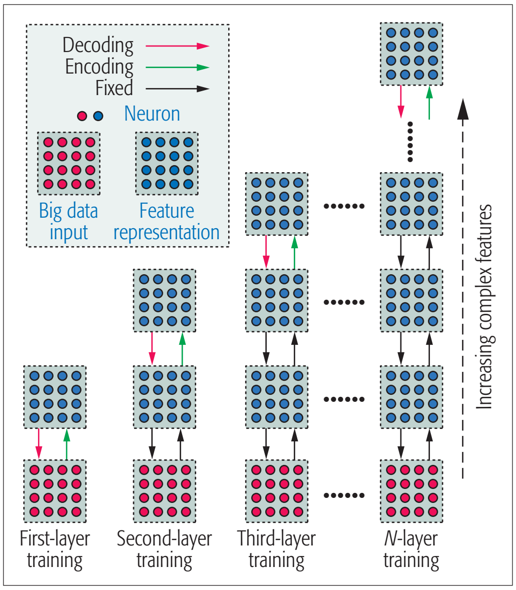

Figure 4. The illustration of critical procedures of the deep-learning-based greedy algorithm.

Discriminative Fine-Tuning: Finally, the model parameters that are initialized in the first step are slightly fine-tuned using the available set of labeled data. In such a manner, the greedy deep learning model can solve the problem at hand.

The deep learning model in big data analysis is slow and old. The Spark-based framework is a novel tool that can speed up big data processing by slicing the data into many partitions. Each elastic distributed dataset can contain a partition. The dataset through Spark offers an abstraction for data distribution. The dataset of Spark can support fault-tolerant executions and recover operations at worker nodes. The novel framework consists of a Spark master and some workers. The master initializes an instance of the Spark driver. Then the execution of many partial models can be managed by the Spark driver in a group of Spark workers. At each iteration of the deep learning algorithm, a worker node learns a partial deep model of big data on a small partition. Next, the computed parameters are sent back to the master node. Then the master node can rebuild a master deep model through delivering the computed partial models of all executor nodes on average.

Deep learning solves the values and varieties of big data issues. Spark tackles the volume, velocity, and volatility aspects of big data. Their contributions to our architecture can be summarized as follows. First, the big data analysis by deep learning helps understand the raw big data; that is, it reveals the values of big data. Second, deep learning enables learning from multi-modal data distribution for the varieties of big data. Third, the Spark-based framework speeds up the big data decision making.

Simulation Results

In smart cities, although the application/service content can be delivered to end users, the quality of applications/services is determined by the precise data. Therefore, we selected machine learning algorithms to process the big data for our proposed architecture so as to extract the precise data. Here, the precise data refers to the useful data that meet end users' demands. In this section, performance evaluation is given to quantitatively validate the algorithm of content delivery in a smart city. The evaluation is conducted via a two-step experiment.

In the first step, we implement our algorithms named deep learning (DL), *K*-nearest neighbor (KNN), and support vector machine (SVM) [13] by MATLAB, and evaluate them using a tensflow simulation tool written by python. We adopt this commonly used simulator because it is designed to import a realistic trace as input from all types of data in databases. We also use the NoSQL dataset [14], which is in various formats of raw data, and collected by sensors from different sources in a smart city. We choose 1 TB data to use by the three mentioned algorithms.

In the second step, we compare the performance of the proposed algorithms in terms of their QoE values, which are computed using the aforementioned QoE model under a specific application of video streaming. Different algorithms are associated with the video quality chosen by end users: SVM algorithm with precise P1, KNN algorithm with precise P2, and DL with precise P3. Then the algorithms are executed aiming to collect their QoE measurements.

The efficiency of algorithms in terms of accuracy, precision, and recall from the algorithms for each test case is evaluated using tensflow. Note that the *accuracy* is defined as the ratio of the number of samples correctly sorted by the classifier dividing the total number of samples for a given test dataset. That is, the loss function is 0–1 loss on the test dataset on the accuracy rate. The *precision* is the ratio between the number of relevant documents retrieved and the total number of documents measured by the search system. The *recall* refers to the ratio of the number of documents retrieved compared to the number of relevant documents in the document library. The result can be seen in Fig. 5. We find that the performance in terms of accuracy, precision, and recall of SVM is the worst, while DL exhibits the best. This is why we adopt the DL algorithm in our proposed big data architecture. As shown in Fig. 6, in detail, the averaged QoE of KNN, SVM, and DL is 70, 62, and 80, respectively. The maximum QoE of KNN, SVM, and DL is 62, 51, and 70, respectively. The minimum QoE of KNN, SVM, and DL is 50, 40, and 71, respectively. We can see that the DL algorithm generates the highest QoE value, and the SVM algorithm yields the lowest QoE.

In summary, we can always observe from both Figs. 5 and 6 that the QoE performance of the three algorithms are positively correlated to their efficiency performance in terms of all three metrics (i.e., accuracy, precision, and recall).

Open Issues

We have presented a novel platform for big data processing for smart cities. Some open issues that are worth further study in the future are summarized as follows.

Dynamic Mobile Big Data: Mobile big data is highly dynamic due to the online arrival demands. Therefore, it is highly challenging to model the spreading acquisition and processing of mobile big data in a smart city.

Economics of Big Data: The soul of big data is about extracting meaningful information and patterns from raw data. The extracted information used during decision making aims to enhance existing services [15]. An important research direction is to propose a business model for trading the big data among organizations and parties.

QoE Model: There is still no clear description of a unified QoE model for comprehensive broadcasting in a smart city. How to integrate the QoE model into the adaptive content decision making engine for optimal playback control is another challenging problem.

Green City: Although some technologies have been used to process big data to extract useful information, the process of big data already presents some challenges. New technologies should be developed to handle big data so as to build a green city.

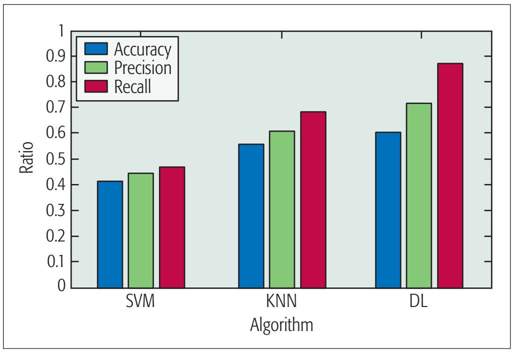

Figure 5. Performance comparison of SVM, KNN, and DL in terms of three metrics: accuracy, precision, and recall.

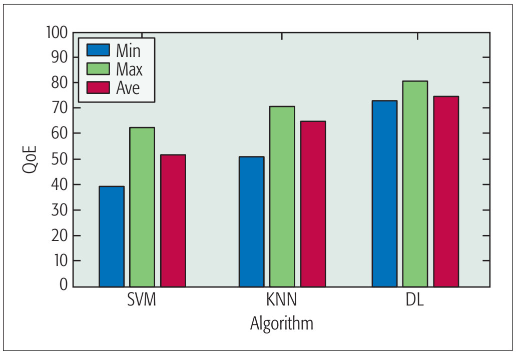

Figure 6. Performance comparison of SVM, KNN, and DL in terms of QoE measurements.

There is still no clear description of a unified QoE model for comprehensive broadcasting in a smart city. How to integrate the QoE model into the adaptive content decision-making engine for optimal playback control is a challenging problem.

Conclusion

In this article, to improve satisfactory QoE for the end users in smart cities, we have proposed a novel big data architecture. As a case study under this architecture, we have devised a deep-learning-based greedy algorithm for the big data services in a smart city to acquire the precise information for end users with satisfactory QoE. Finally, simulation results demonstrate the high QoE performance of the proposed deep-learning-based algorithm.

Acknowledgment

This work is supported by NSFC (61572262, 61533010, 61373135, 61571233, 61532013); the National China 973 Project (2015CB352401); the China Postdoctoral Science Foundation (2017M610252); the China Postdoctoral Science Special Foundation (2017T100297); the Open Research Fund of the Jiangsu Engineering Research Center of Communication and Network Technology, NJUPT; and the National Engineering Research Center of Communications and Networking (Nanjing University of Posts and Telecommunications) (TXKY17014).

References

[1] Z. Su, Q. Xu, and Q. Qi, "Big Data in Mobile Social Networks: A QoE-Oriented Framework," *IEEE Network*, vol. 30, no. 1, Jan./Feb. 2016, pp. 52–57.

[2] W. Rahman, D. Yun, and K. Chung, "A Client Side Buffer Management Algorithm to Improve QoE," *IEEE Trans. Consumer Electronics*, vol. 62, no. 4, Feb. 2016, pp. 371–79.

[3] H. Tran *et al.*, "QoE-Based Server Selection for Content Distribution Networks," *IEEE Trans. Computers*, vol. 63, no. 11, Feb. 2014, pp. 2803–15.

[4] Z. Su and Q. Xu, "Content Distribution over Content-Centric Mobile Social Networks in 5G," *IEEE Commun. Mag.*, vol. 53, no. 6, June 2015, pp. 66–72.

[5] R. Polishetty, M. Roopaei, and P. Rad, "A Next-Generation Secure Cloud-Based Deep Learning License Plate Recognition for Smart Cities," *Proc. 15th IEEE Int'l. Conf. Machine Learning and Applications*, Anaheim, CA, 2017.

[6] K. Wang *et al.*, "Wireless Big Data Computing in Smart Grid," *IEEE Wireless Commun.*, vol. 24, no. 2, Apr. 2017, pp. 58–64.

[7] K. Wang *et al.*, "Mobile Big Data Fault-Tolerant Processing for eHealth Networks," *IEEE Network*, vol. 30, no. 1, Jan./Feb. 2016, pp. 36–42.

[8] K. Wang *et al.*, "Real-Time Load Reduction in Multimedia Big Data for Mobile Internet," *ACM Trans. Multimedia Computing, Commun. and Applications*, vol. 26, no. 5, article 76, Oct. 2016.

[9] G. Jia *et al.*, "Dynamic Adaptive Replacement Policy in Shared Last-Level Cache of DRAM/PCM Hybrid Memory for Big Data Storage," *IEEE Trans. Industrial Informatics*, vol. 13, no. 4, Aug. 2017, pp. 1951–60.

[10] K. Mitra, A. Zaslavsky, and C. Ahlund, "Context-Aware QoE Modelling, Measurement, and Prediction in Mobile Computing Systems," *IEEE Trans. Mobile Computing*, vol. 14, no. 5, Dec. 2015, pp. 920–36.

[11] M. Alsheikh *et al.*, "Mobile Big Data Analytics Using Deep Learning and Apache Spark," *IEEE Network*, vol. 30, no. 3, May/June 2016, pp. 22–29.

[12] R. Sarikaya, G. Hinton, and A. Deoras, "Application of Deep Belief Networks for Natural Language Understanding," *IEEE Trans. Audio, Speech and Language Processing*, vol. 22, no. 4, Feb. 2014, pp. 778–84.

[13] J. Hou *et al.*, "Feature Combination and the KNN Framework in Object Classification," *IEEE Trans. Neural Networks and Learning Systems*, vol. 27, no. 6, Aug. 2016, pp. 1368–78.

[14] L. Likforman-Sulem *et al.*, "EMOTHAW: A Novel Database for Emotional State Recognition from Handwriting and Drawing," *IEEE Trans. Human-Machine Systems*, vol. 47, no. 2, Jan. 2017, pp. 273–84.

[15] K. Wang et al., "LDPA: A Local Data Processing Architecture in Ambient Assisted Living Communications," *IEEE Commun. Mag.*, vol. 53, no. 1, Jan. 2015, pp. 56–63.

Biographies

Xiaoming He is a postgraduate student in the School of Internet of Things, Nanjing University of Posts and Telecommunications, China. His current research interests include vehicle-to-grid, big data, machine learning, and content-centric networking.

Kun Wang [SM'17] received his Ph.D. from Nanjing University of Posts and Telecommunications in 2009. From 2013 to 2015, he was a postdoctoral fellow in the Electrical Engineering Department, University of California Los Angeles. He is currently a research fellow at Hong Kong Polytechnic University and also a full professor in Nanjing University of Posts and Telecommunications. His current research interests are mainly in the area of big data, wireless communications and networking, smart grid, energy Internet, and information security technologies.

Huawei Huang [M'13] received his Ph.D. in computer science from the University of Aizu, Japan. His research interests mainly include networking optimization and algorithm design/analysis, particularly in the fields of software-defined networking, network functions virtualization, and wireless networks. He is a JSPS Research Fellow.

Bo Liu [M'10] received his Ph.D. in 2010 from Shanghai Jiao Tong University, China. Since 2010, he has been an assistant researcher with the Electrical Engineering Department, Shanghai Jiao Tong University. Since November 2014, he has been a postdoctoral research fellow at Deakin University, Melbourne, Australia. His research interests include wireless communications, networking protocols, and privacy issues in wireless networks.