

# Smart Infotainment Service Management for Public Transportation

<sup>1</sup>Pyae Sone Aung, <sup>2</sup>Kyi Thar, <sup>3</sup>Anselme Ndikumana, <sup>4</sup>Choong Seon Hong  
Department of Computer Science and Engineering, Kyung Hee University  
Yongin, 446-701 Korea  
{<sup>1</sup>pyaesoneaung, <sup>2</sup>kyithar, <sup>3</sup>anselme, <sup>4</sup>cshong}@khu.ac.kr

## Abstract

In this paper, smart infotainment service for caching video contents is proposed to reduce the network redundancy and video accessing delay. The contents are temporarily stored inside the local cache storage proactively based on the contents' popularity and the challenging issue is to determine which popular contents to store. Here, branch and bound prediction scheme is proposed to generate deep learning models for calculating the contents' popularities. To implement the proposed prediction scheme, we use the MovieLens dataset to train the learning scheme and use machine learning libraries such as Keras and Tensorflow. The simulation results of our proposed approach has better performance than other schemes under comparison in prediction content popularity.

## 1. Introduction

Cisco Visual Networking Index (CNI) mentions that nowadays, most of the Internet traffic is consumed by video streaming in wireless devices [1]. Also, in the near future, every transportation bus will be implemented with smart displays supported for individual seats which support infotainment systems that provide passenger centric services such as streaming videos, advertisement, and safety notifications based on passenger information (e.g., passenger's emotion, gender information, etc...). With the emerging technology of Multi-access Edge Computing (MEC), a real time data analytics can be executed and popular contents can be cached at the edge network nodes such as base stations, buses [2].

Among these services, this paper is focused on demand video service in which the videos are recommended to the passengers, according to their current detected emotions. Due to the page limitation, in this paper, we will only focus on predicting the content's popularity and caching service based on the user's movie watching patterns. Assume that passengers only watch the movies from the personalized recommended list.

Since, buses are high in mobility and poor-quality wireless links, some popular video

contents need to be cached at the bus for faster video access before requested by passengers. So, to solve the aforementioned issues, in this paper, we proposed the smart infotainment service based on one of the deep learning based content's popularity prediction methods and the content caching system at the edge device which is installed at the bus. Our proposal contributions are as followings:

- The main challenging issue to apply the deep learning into contents' popularity prediction scheme is that we need to find the best-deep learning model among various types such as recurrent neural network (RNN), and multi-layer perceptron (MLP). Tuning hyper parameters is also another challenging problem. To solve the challenging issues, we propose the randomized searching algorithm with branch and bound method.
- The proposed system is trained with the dataset called MovieLens [3] with the usage of the machine learning libraries such as Keras and Tensorflow.

So, as a result, time for video accessing delay and the number of retrieving recurrent videos from the content servers is fairly reduced with proposed content caching schemes compared

to the without caching scheme and the probabilistic caching scheme.

## 2. System Model

The proposed system model is shown in Fig. 1, where Cloud Data Center handles for searching the suitable deep learning architecture and constructing the best-suited model for content's popularity prediction. Then, the constructed deep learning model is sent to the smart infotainment system at the bus to track the content's popularity. Assume that, the content server stores all of the contents requested by passengers. The infotainment system consists of i) Edge Content Manager, ii) Prediction Module, iii) Content Storage, iv) Request Handler, v) Request Log and vi) Smart Display on the passenger's seats. Edge Content Manager makes a decision to download the popular contents and make a cache decision (the decision whether to store the contents and delete the old ones). The prediction module is responsible to predict the contents' popularity based on the prediction model sent from the cloud data center. Content Storage temporarily caches the popular contents. Request Handler makes a decision to route the content requests to the base station when the requested contents are not located in the Content Storage.

The overview processes of the proposed scheme are as follows. With the requests arrival from users, the request handler first searches for the contents inside the local cache content storage. If the cache content is not found, the content will be retrieved from the content server through the base station by the Edge Content Manager (ECM). At the same time, the request handler also stores the user requests inside the request log. Here, the proactive caching scheme is considered whether the cache decision stores the downloaded contents inside the local cache storage or not, and to delete the old ones. At every time  $t$ , the prediction module uses deep learning model to predict the contents' popularity, based on the request counts from the users and pre-store the caches inside the

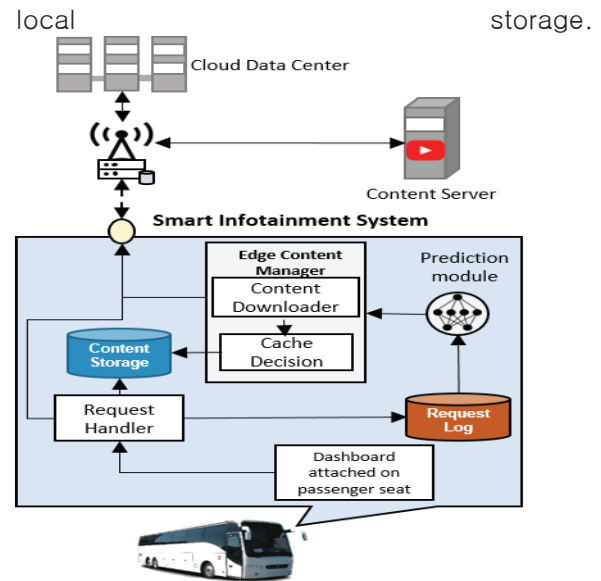


Figure 1 System Model

## 3. Prediction of the popularity of content

We find the most suitable models for predicting the popularity class and the request counts among the three models (LSTM, CNN and CRNN) with numerous configurations for the hyper parameters, for example, the number of layers. Each learning model can have a huge number of configuration combinations. To search the best-suited models, we can use the two methods. The former one is the grid-search and the latter one is called random-search with branch and bound. How grid-search works is that it finds the best model by the increment of hyper parameter values in sequential combinations. For the random-search, it uses random combinations of the hyper parameters in order to reduce the search space to find the best model. To reduce complicated computation, we choose the latter one and the hyper parameters are randomly configured in order to create the models.

## 4. Performance Evaluation

We use the MovieLens dataset to evaluate the performance of our proposed prediction and caching scheme. This dataset contains 26,000,000 ratings given to 45,000 movies and those are rated by 270,000 viewers. We assume the rating counts is equal to request counts for the movies and the time for users

rating movies are equivalent to the request arrival times.

Due to page limitation, we only consider the best 10 class prediction validation accuracy among the 50 deep learning models chosen randomly in Fig.2. From Fig. 2, it is clear that the configuration of CLSTM <40>, Dense <250>, (i.e., CLSTM model with 40 cells in CLSTM layer and 250 neurons in the dense layer), provides the best validation accuracy and thus is chosen using class prediction Algorithm 1. Fig.3 shows the comparison of the probability of cache hit between our proposed system, randomized caching and no caching in which the higher cache hit probability, the better performance. With increment in the size of the cache, the probability of a cache hit also escalates due to more storage capacity to store the predicted contents. This proves that our proposed model provides better results than other schemes in prediction of the most popular contents and caching them proactively.

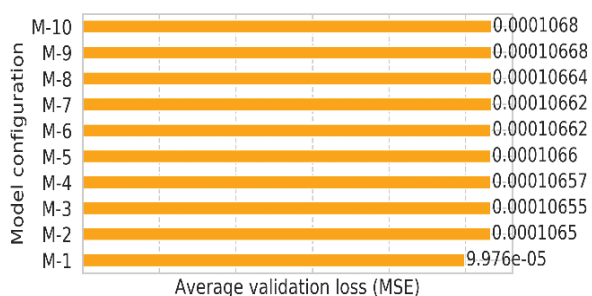


Fig. 2. Average validation loss of 10 among 50 deep learning models randomly chosen

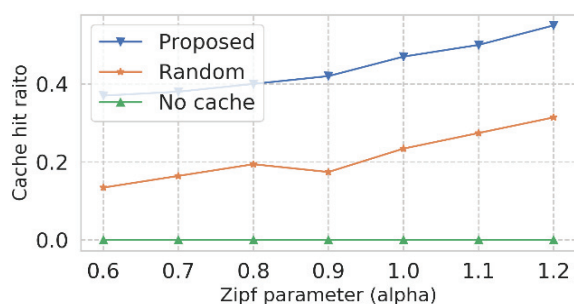


Fig. 3. Cache hit comparison among Proposed Scheme, Probabilistic Caching, and No Caching

**Algorithm 1** Model generating and caching process

**Input :** Dataset  
**Output:** Cache Decision

- 1 Search the best-suited deep learning model for content popularity prediction among CNN, LSTM and CLSTM architecture with branch and bound methods
- 2 Choose the searched model which has lowest validation loss
- 3 Train the chosen model at the data center
- 4 Distribute the trained model to infotainment system at the buses
- 5 Every time  $t$ , predict the content's popularity and download the contents from the server and store at the Content Storage

**5. Conclusion**

In this paper, content popularity prediction based on deep learning and caching scheme are used to proactively store the video contents at storage attached with the bus to improve the cache hit probability. Also, we proposed the deep-learning model searching scheme based on the branch and bound method, to find the popularity prediction model. As a future work, we will continue to improve in generating deep learning models with other suitable methods.

**Acknowledgement**

This work was partially supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2015-0-00557, Resilient/Fault-Tolerant Autonomic Networking Based on Physicality, Relationship and Service Semantic of IoT Devices) and by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (NRF-2016R1D1A1B01015320). \*Dr. CS Hong is the corresponding author

**References**

- [1] Accessed: Feb. 7, 2019. [Online]. Available:<http://www.cisco.com/c/en/us/solutions/service-provider/visualnetworking-index-vni/index.html>
- [2] Ndikumana, A., Tran, N. H., & Hong, C. S. (2018). Deep Learning Based Caching for Self-Driving Car in Multi-access Edge Computing. arXiv preprint arXiv:1810.01548.
- [3]<https://grouplens.org/datasets/movielens/>