MLP를 사용하여 DASH 데이터 세트의 세그먼트 분류

Linh Van Ma*, Van Quan Nguyen*, 김진술***

Classify Segments from DASH Dataset Using MLP

Linh Van Ma*, Van Quan Nguyen**, Jinsul Kim***

요 약

HTTP (DASH)를 통한 동적 적응형 스트리밍은 기존 HTTP 웹 서버에서 전달되는 인터넷을 통해 고품질의 미디어 콘텐츠 스트리밍을 가능하게 하는 적응형 bitrate 스트리밍 기술입니다. 본 연구에서는 다층 퍼셉트론 (MLP) 분류 기술을 사용하여 DASH 데이터세트를 분류하는 학습 알고리즘을 살펴 본다. 결과적으로, NS3 (Network Simulator 3)를 사용하여 분류 결과를 시뮬레이션에 적용한다. 이러한 과정은 우선적으로 교육 데이터세트에 적용된 다음, 그 결과가 테스트 데이터세트에 사용된다.

Abstract

Dynamic Adaptive Streaming over HTTP (DASH) is an adaptive bitrate streaming technique that enables high-quality streaming of media content over the Internet delivered from conventional HTTP web servers. In this research, we take a look at a learning algorithm which classifies DASH dataset using multilayer perceptron (MLP) classification technique. Consequently, we apply the classification result into a simulation by using NS3 (Network Simulator 3). The process firstly applies to the training dataset, and then the result is used in the testing dataset.

Key words
DASH, Multilayer Perceptron, Deep Learning, Classification, NS3

Ⅰ. Introduction

Dynamic Adaptive Streaming over HTTP (DASH) [1] is an adaptive bitrate streaming technique that enables high-quality streaming of media content over the Internet delivered from conventional HTTP web servers. First, the content is made available at a variety of different bit rates.Secondly, while the content is being played back by an MPEG-DASH client, the client automatically selects the segment with the highest bit rate possible that can be downloaded in time for playback without causing stalls or re-buffering events in the playback. An adaptive algorithm entitled "FDASH: A Fuzzy-Based

* ** *** School of Electronics and Computer Engineering, Chonnam National University, Gwangju, Korea

MPEG/DASH Adaptation Algorithm" [2] has been proposed recently can efficiently choose the highest bit rate possible. The overview of the article is shown in Figure 1. We have two inputs: 1) Buffer Differential is the differential of the buffering time input we need to describe the behavior of the rate between subsequent buffering times; 2) Buffer Estimate (second): is the current estimation of the buffer in the client. The output is the one of the given bitrates (bits per second) in the right. With the simulation data gotten from NS3 (Network Simulator 3) for FDASH, classify segments in the dataset (training, testing dataset)

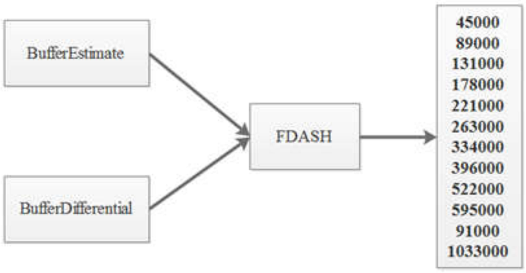

Figure 1: Overview of FDASH

II. System Overview and Experiment

The FDASH algorithm has been implemented on NS3. We simulate the algorithm in one day with three users in the network link rate (1000Kbps). Two dataset gotten from two users (232684x2 records) is used to train and build the model, and data set obtain from the rest user (232684 records) is used to test the model. Subsequently, we utilize Waikato Environment for Knowledge Analysis (Weka) [3] which is a suite of machine learning software written in Java, to solve the above-defined problem. In the demonstration, we set learning rate equal to 0.3, momentum to 0.2, and epoch to 300. The input of the MLP (Multiple Layer Perceptron) is Buffer Estimate and Buffer Differental as described in the problem above. The output is the set of video representation {45000, 89000, 131000, 178000, 221000, 263000, 334000, 396000, 522000, 595000, 91000, 1033000} where we can only choose one representation at once time.

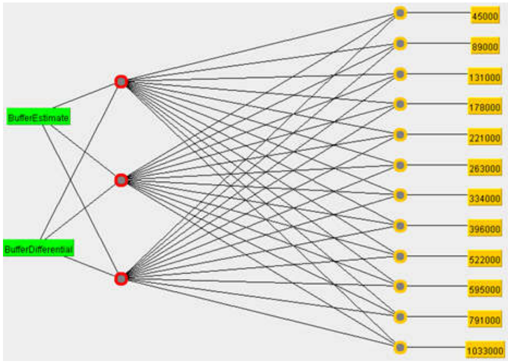

Figure 2: One hidden layer with three perceptron

First, we run the program with one hidden layer and three perceptrons as shown in Figure 2. The result is shown in Table 1. After training, the model is applied to test data which has 232684 instances. It correctly classified almost 91% of the total instances with root mean squared error is 0.1015.

Table 1: classification with one hidden layer and three perceptrons

Correctly Classified Instances	210890	90.6336 %
Incorrectly Classified Instances	21794	9.3664 %
Root mean squared error	0.1015	
Total Number of Instances	232684	

Secondly, we run the program with one hidden layer and eight perceptrons. After training, the model is applied to test data which has 232684 instances. It correctly classified almost 91% of the total instances with root mean squared error is 0.1025.

Finally, we set different learning rates and execute the classification program and got the comparison of convergence of the MLP as shown in Figure 3. It shows that the learning rate (0.2) leading to faster convergence than the other two.

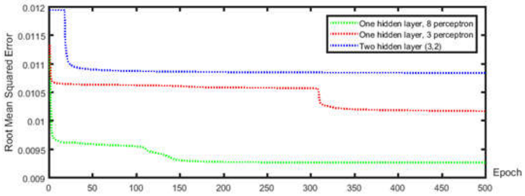

Figure 3: Comparison of RMSE of the three MLP

III. Conclusion

In this paper, we have a glance demonstration at the learning algorithm (MLP) in adaptive streaming for the FDASH algorithm. The result shows that, it can classify 90% of the testing dataset upon a training. In the future research, we are going to apply many classification algorithms in various proposed adaptive streaming to form a general adaptive streaming algorithm.

Acknowledgments

This research was supported by the IT R&D program of MSIT (Ministry of Science and ICT), Korea / NIPA (National IT Industry Promotion Agency) 12221-14-1001, Next Generation Network Computing Platform Testbed. Besides, this research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science, and Technology (MEST) (Grant No. NRF-2017R1D1A1B03034429).

References

[1] Ma, Linh Van, Jaehyung Park, Jiseung Nam, HoYong Ryu, and Jinsul Kim, A Fuzzy-Based Adaptive Streaming Algorithm for Reducing Entropy Rate of DASH Bitrate Fluctuation to Improve Mobile Quality of Service, Entropy Vol. 19, No. 9, 2017, pages. 477-489.

[2] Vergados, Dimitrios J., Angelos Michalas, Aggeliki Sgora, Dimitrios D. Vergados, and Periklis Chatzimisios, Fdash: a fuzzy-based MPEG/DASH adaptation algorithm, IEEE Systems Journal Vol. 10, No.2, 2016, pages. 859-868.

[3] Kalmegh, Sushilkumar, Analysis of WEKA data mining algorithm REPTree, Simple CART and RandomTree for classification of Indian news, Int. J. Innov. Sci. Eng. Technol Vol. 2, No. 2, 2015, pages. 438-446.