FERA: A Caching Scheme in CCN using File-Extension and Regression Analysis

Jin Won Lee, Choong Seon Hong
Department of Computer Science and Engineering
Kyung Hee University, 446-701
The Republic of Korea
Email: {notwonz, cshong}@khu.ac.kr

***Abstract*— Content-Centric Networking(CCN) is emerging technology for the future network infrastructure. In CCN, CCN routers support cache space called Content Store(CS), which can store passing contents. In this paper, we define two representative problems that can occur in CS of CCN routers, namely content search problem and cache replacement problem, and propose FERA(File Extension and Regression Analysis based Caching Scheme) which can solve these problems. In FERA, two approaches are proposed. Specifically, the first proposal is to divide CS into four types of CSs based on file extensions. Thereafter, when the user sends a request including the content name, the CCN router extracts and identifies the file extension from the content name, and transmits the user's request to the corresponding CS. After being transmitted, the user request is processed in the corresponding CS. The second proposal is to predict the content that is continuously decreasing in popularity based on regression analysis which is one of the machine learning algorithms and evicts that content. Finally, we verify the proposed scheme using CCN simulator. The simulation results show that the proposed scheme improves the cache hit ratio and the cache hit distance over the LRU.**

Keywords—Content-Centric Networking(CCN), Content Store, File Extension, Cache Replacement Policy, Regression Analysis.

I. Introduction

The current Internet uses TCP / IP communications based on IP addresses developed to provide services to non-mobile users in the 1970s. In addition, Internet traffic has exploded in recent years due to advances in wireless network technologies and smartphone and application technologies. In this situation, many users have the characteristic of requiring the same service and contents of certain famous web pages such as YouTube and Facebook. However, TCP / IP communication requires multiple users to access one IP address in order to obtain the same contents, which can cause problems such as bottlenecks in Figure 1, unnecessary location search, and repeated transmission. Therefore, the need for a new future network structure has emerged and Content-Centric Networking (CCN) has been proposed by Van Jacobson [1].

The CCN is characterized by the use of content names instead of IP addresses, and the content store (CS), which can store all the content that passes between the content servers and users. CS allows users to quickly acquire frequently requested content from a nearby router and solve problems such as TCP/IP communication bottlenecks as shown in Figure 2.

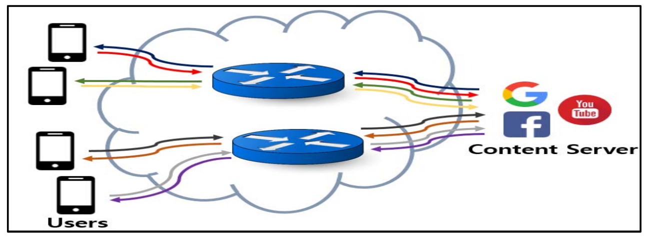

Figure 1. Bottleneck in TCP / IP communication

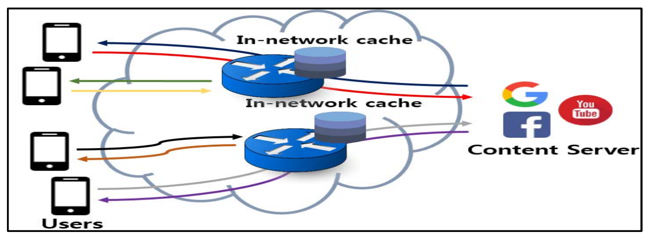

Figure 2. CCN with Bottleneck resolved

In Section III, we define content search problem and cache replacement problem that can occur in CS of CCN. In Section IV, we propose FERA(File Extension and Regression Analysis based Caching Scheme) and explain its operation. In Section V, we compare the cache hit ratio and cache hit distance of the proposed CCN with the LRU using ccnSim which is a CCN simulator based on OMNET ++.

II. Related works

A. Content-Centric Networking

TCP / IP communication uses IP addresses to acquire content, while CCN uses content names to acquire content. CCN uses two types of Packets: Interest Packet used when requesting content and Data Packet used when responding to requests and delivering content. In addition, each CCN router consists of three kinds of caches: Content Store(CS) for storing contents, Pending Interest Table(PIT) for storing a record of an Interest Packet, and Forwarding Information Base(FIB) for forwarding Interest Packets.

The user sends the Interest Packet containing the name of the content to request, and the CCN router that receives the Interest Packet searches the CS based on the name of the requested content. If there is matched content in the CS, it sends a Data Packet containing the matched content to the requested interface. If there is no matched content in the CS, it sends an Interest Packet to the PIT and FIB to acquire the content from the upper router or content provider(server)[2].

Since CCN supports caches such as CS, PIT, FIB, users can acquire content from an intermediate node without accessing certain IP address like content provider or server. Also, since CCN packets have only information about contents(e.g., content name), they can defend against attacks targeting specific host such as DDoS attacks.

B. Cache Replacement Policy

The CCN router stores all the content going through the router in the CS. At this time, if there is no more space to store the new content, the CCN router selects the content to be evicted from the CS and replaces that content with the content to be newly stored. This called a cache replacement policy.

The cache replacement policy stores frequently requested content(popular content) in the CS to reduce network bandwidth, server load and allows users to quickly acquire popular content. Currently, the most common cache replacement policies are LRU, LFU, and FIFO. More research is underway to find cache replacement policies that are more efficient than the algorithms mentioned above[3], [4], [5].

III. Problem Formulation

In this section, we define inefficient content searching, unfair caching methods and cache replacement problems that are problems of existing CCN to solve in this paper.

A. Content Searching Problem

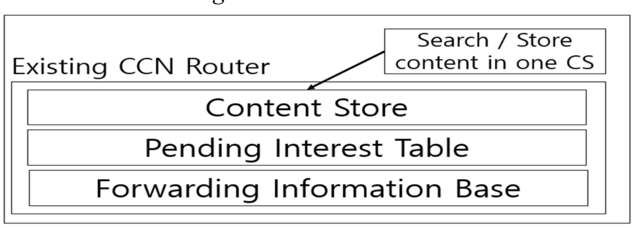

Figure 3: Structure of existing CCN Router

In the existing CCN, when a CCN router receives a content request, it searches for content in one CS. This process has two problems. The first problem is that the search takes a long time. Because they search until they find matched content based on the content name. The second problem arises when a content replacement policy performed with different extensions. Content with different extensions means different purposes of use, and the frequency with which users request content is different according to the characteristics of extensions. For example, suppose you have video content and image content. Since the video content has a large capacity in terms of characteristics, the number of user's requests is small during a unit time, but image content is requested by a plurality of users in a short cycle. If video and image content are replaced at the time and frequency of request, unfair replacement can occur, because CCN basically uses LRU and LFU cache replacement policy. Therefore, in Section IV, we divide CS by file extension to reduce content search time. Also, we propose an algorithm to perform fair replacement by exchanging similar extensions.

B. Cache Replacement Problem

In the existing CCN, when the CS is full, the cache replacement policy such as the LRU and the LFU is performed to select and evict the content. The criterion for selecting the content to be evicted by the LRU, LFU is the time and the number of times the content was used. In this process, there may be content that is popular but temporarily reduces user requests. If the user's request is temporarily reduced, that content is evicted from the CS due to the cache replacement policy of the LRU and LFU. However, since the evicted content is popular, the user's request is increased again, and the probability of being stored again in the CS is high. An unnecessary operation is additionally required in order to select contents to be emitted and to store the evicted contents again in the CS. Additional operations can cause network performance degradation and delay the content delivery of CCN routers. Therefore, in Section IV, we propose a cache replacement policy that predicts the flow of popularity, not the instant popularity of content, using regression analysis of machine learning algorithms.

IV. FERA: File Extension and Regression Analysis based Caching Scheme

In this section, we propose a FERA to solve each problem of CCN mentioned in Section III-A, III-B.

A. Content Store divided by File Extension

Table 1. File Extension Classification

Video-CS	Audio-CS	Image-CS	ETC-CS
.avi	.acc	.jpg	.apk
.mp4	.mp3	.jpeg	.exe
.mpg	.wav	.png	.alz
.mpeg	.ogg	.dib	.app
.mpg	.mp2	.tiff	.pdf
.fli	.m3u	.bmp	.html
.flm	.cda	.img	.zip

Figure 4 shows the proposed structure of CCN router in this paper. In the proposed CCN router, the CS is divided into 4 types(Video-CS, Audio-CS, Image-CS, and ETC-CS) according to Table 1, and the CCN router identifies the file extension when receiving the CCN packet, and searches and stores the content in the corresponding CS. The proposed architecture enables fair content replacement by replacing contents with similar file extensions and fast content searching.

The operation of the proposed CCN router is as follows. First, when the CCN router receives a content search or store request, the CCN router extracts the content name and file extension from the request. Then, the CCN router selects the CS to perform the work based on the file extension and searches the content based on the content name.

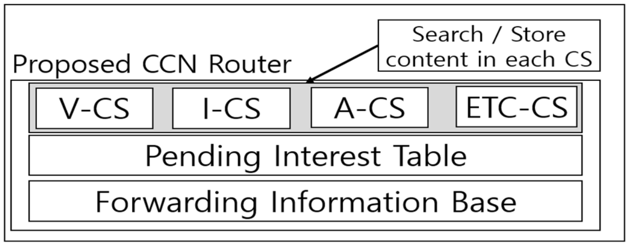

Figure 4. Structure of proposed CCN Router

If there is matched content requested by the client in the CS, the content is delivered to the client. However, if there is no matched content in the CS, the PIT and FIB are used to forward the request to the upper router. After receiving the content from the upper router, CCN router stores it in the CS and delivers it to the client. Since the proposed structure searches by similar file extensions, it enables faster and more efficient search than existing CCN which compares all content names. Also, when the CS is full and the content needs to be replaced, the client's cache hit ratio is improved because the proposed CCN router replaces content between similar file extensions.

B. Regression Analysis based Cache Replacement Policy

Table 2. Structure of proposed Content Store

Proposed Content Store		
Content Name	History Popularity	Current Popularity
Content 1	$H_1[t-1]$	$C_1[t]$
Content 2	$H_2[t-1]$	$C_2[t]$
…	…	…
Content n	$H_n[t-1]$	$C_n[t]$

Table 2 shows the internal structure of each CS proposed in Figure 4. In the proposed CS structure, besides the content and the name, the popularity of the content is stored together. Popularity consists of history popularity and current popularity. The formula for calculating history and current popularity is as follows.

In formula (1) and (2), t represents the time when the cache replacement policy is performed and i represents the i-th specific content among all the contents in the CS. (3), which is the change amount of the past present popularity, can be obtained through (1) and (2). The regression analysis algorithm learns the change of (3) over time and determines the content be evicted at the time t when cache replacement is to be performed. When selecting the content, the content with the lowest slope in the regression equation is determined. Definition, independent and dependent variables of regression analysis are defined in Table 3.

$$H_i[t-1] = \frac{\sum_{t-2}^{t-1} Req(i)}{\sum_{t-2}^{t-1} Req(total)} \quad (1)$$

$$C_i[t] = \frac{\sum_{t-1}^{t} Req(i)}{\sum_{t-1}^{t} Req(total)} \quad (2)$$

$$\Delta P_i[t] = C_i[t] - H_i[t-1] \quad (3)$$

Table 3. Variable Description

VARIABLE	DESCRIPTION
$\Delta \hat{\boldsymbol{P}}_{\boldsymbol{i}}[\boldsymbol{t}] = \boldsymbol{b_0} + \boldsymbol{b_1 t}$	Regression model
$\boldsymbol{b_1} = \frac{\sum(\boldsymbol{t}-\bar{\boldsymbol{t}})(\Delta \boldsymbol{P_i}[\boldsymbol{t}] - \Delta \bar{\boldsymbol{P}}_i[\boldsymbol{t}])}{\sum(\boldsymbol{t}-\bar{\boldsymbol{t}})^2}$	Slope of regression line
$\boldsymbol{b_0} = \Delta \bar{\boldsymbol{P}}_i[\boldsymbol{t}] - \boldsymbol{b_1}\bar{\boldsymbol{t}}$	Intercept
$\bar{\boldsymbol{t}}$	Average of t
$\Delta \bar{\boldsymbol{P}}_{\boldsymbol{i}}[\boldsymbol{t}]$	Average of $\Delta P_i[t]$

Algorithm 1 shows an operation of regression analysis algorithm for predicting content that has a steady decrease in popularity among the contents existing in the CS and is least likely to be stored again in the CS in the future.

Algorithm 1 Proposed Cache Replacement Policy

```
PREMISE: Regression Analysis algorithm always calculates
         content popularities.
INPUT : Req(j) (Request for content j)
if j is in CS then
  Return j to client
else
  Forward Req(j) to upper router and Get content j
  if CS is full then
    Select content k which has the lowest b1
    Evict content k and Cache content j
  else
    Cache content j
  end if
end if
Return j to client
```

V. PERFORMANCE EVALUATION

In this section, we present simulation setup and simulation results of our proposed scheme.

A. Simulation Setup

We used CCN Simulator ccnSim(0.3) to compare cache hit ratio and cache hit distance with LRU, which is the existing CCN's cache replacement policy. The formula for calculating the cache hit ratio(4) is as follows and the simulation environment is shown in Table 4.

$$Cache\ Hit\ Ratio = \frac{Cache\ Hit}{Cache\ Hit + Cache\ Miss} \quad (4)$$

Table 4. Simulation Environment

Parameter	**Value**	**Parameter**	**Value**
Number of Users	50	Content Size	1
Cache Size	30~70%	Number of CCN routers	10
Zipf parameter	0.5~1.2	Chunks per Content	1
Number of Contents	10,000	Number of Repositories	1

The packet forwarding strategy and the cache decision algorithm are applied to NRR1(Nearest Neighbor Routing) and LCE(Leave Copy Everywhere), respectively, and compared with the LRU, a cache replacement technique of the existing CCN, to show the cache hit ratio improvement of the proposed scheme.

B. Simulation Results

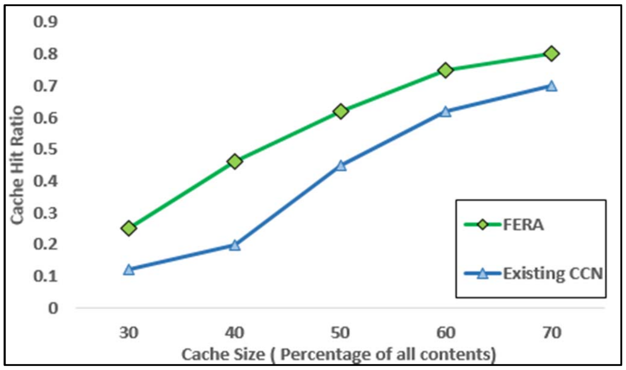

Figure 5. Comparison of cache hit ratio with LRU

Figure 5 shows the comparison of the cache hit ratio between FERA and the LRU. When the cache size ratio increases from 30% to 70% of the total content size, the cache hit ratio of the FERA and the LRU increases gradually because the content acceptance rate of the CS increases. However, since the FERA cache policy ensures fair cache and fair cache replacement by storing each content in the CS divided by the file extension, the cache hit ratio of the user is greatly increased compared to the LRU.

Figure 6 shows the result of comparing the cache hit distance between FERA and the LRU. Cache hit distance indicates how close the CCN router receives the content when the user requests the content. Because FERA predicts the amount of change in popularity of each content and evicts unnecessary content, routers are made up of substantially popular content. Therefore, it has a lower cache hit distance than the existing CCN LRU policy. As the Zipf parameter increases from 0.5 to 1.2, certain popular contents occupy most of the user's requests, so a small number of specific popular content is stored close to the user, reducing the cache hit distance.

VI. Conclusion

Content-Centric Networking(CCN) is one of the future network paradigms. In the CCN, CCN router has a cache space called a content store. In this paper, we proposed FERA (File Extension and Regression Analysis based Caching Scheme) which can solve content search problem and cache replacement problem that can occur in the content store of CCN.

In order to solve the content search problem, we divide content store into four kinds based on the file extension and to solve the cache replacement problem, we designed an algorithm that predicts and evicts the content with steadily decreasing

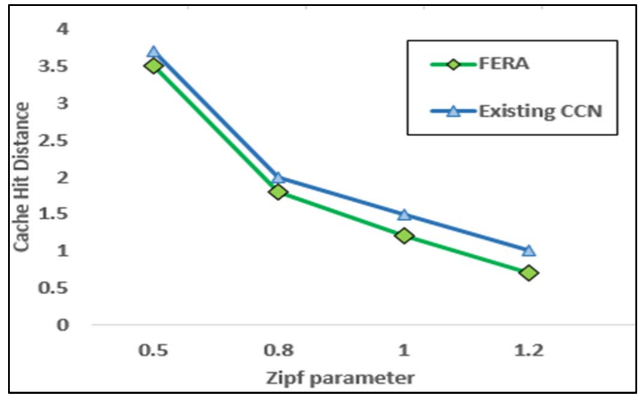

Figure 6. Comparison of cache hit distance with LRU

popularity by using regression analysis among machine learning algorithms.

We simulated using ccnSim and compared the improvement of a cache hit ratio and cache hit distance with LRU. The future research plan is to allocate the more meaningful size of each CS divided based on the file extension. When allocating the cache size, instead of counting the number of requests, we will use a machine learning algorithm to predict the user's request pattern and allocate each CS size based on the prediction.

Acknowledgment

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (B0101-16-0033, Research and Development of 5G Mobile Communications Technologies using CCN-based Multi-dimensional Scalability) *Dr. CS Hong is the corresponding author

References

[1] Van Jacobson, Diana K Smetters, James D Thornton, Michael F Plass, Nicholas H briggs, Rebecca L Braynard, “Networking named content,” In Proceedings of the 5th International conference on Emerging networking experiments and technologies, pp. 1-12. ACM, 2009

[2] Bing Li, Maode Ma, Xinbin Yang, “Advanced Perceptive Forwarding in Content-Centric Networks,” IEEE Access, vol. 5, pp. 4595-4605. IEEE, 2017.

[3] Jang-Ping Sheu, Yen-Cheng Chuo, “Wildcard Rules Caching and Cache Replacement Algorithms in Software-Defined Networking,” IEEE Transactions on Network and Service Management, vol. 13, pp. 19-29. IEEE, 2016.

[4] Mazen Kharbutli, Rami Sheikh, “LACS: A Locality-Aware Cost-Sensitive Cache Replacement Algorithm,” IEEE Transactions on Computers, vol. 63, pp. 1975-1987. IEEE, 2014.

[5] Emira Ben abdelkrim, Mohammad A. Salahuddin, Halima Elbiaze, Roch Glitho, “A Hybrid Regression Model for Video Popularity-based Cache Replacement in Content Delivery Networks,” IEEE Global Communications Conference(GLOBECOM), pp. 1-7. IEEE, 2016.