Semantic Edge Caching and Prefetching in 5G

Can Mehteroglu
REHIS Division
Aselsan Incorporation
Ankara, Turkey
Email: cmehteroglu@aselsan.com.tr

Yunus Durmus
Accenture
The Netherlands
Email: yunus.durmus@accenture.com

Ertan Onur
Department of Computer Engineering
Middle East Technical University
Ankara, Turkey
Email: eronur@metu.edu.tr

***Abstract*—Recent popularity of mobile devices increased the demand for mobile network services and applications that require minimal delay. 5G mobile networks are expected to provide much lesser delay than the present mobile networks. One of the conventional ways for decreasing the latency is caching the content closer to the end user. However, currently deployed methods are not effective enough. In this work-in-progress paper, we propose a new astute caching strategy that is able to smartly predict subsequent user requests and prefetch necessary contents to remarkably decrease the end-to-end latency in 5G systems. We employ semantic inference by mobile edge computing, deduce what the end-user may request in the sequel and prefetch the content.**

***Index Terms*—Caching; 5G mobile networks; Mobile edge computing**

I. Introduction

Smart mobile devices are capable of internet browsing, location tracking, streaming high definition videos and many more, at any time and place. Since the penetration of these devices are high in the societies, we regard them as a huge technological success. However, the expectations of the consumers grow steadily in terms of both data rates and end-to-end latency of mobile services. In 2G systems, the focus was on the voice communication and its requirements were driven by the human audible delay constraints [1]. After that, multimedia applications became prominent and they require less than 10 miliseconds delay because human eye is more sensitive than the ear. When touch screens are considered, the delay requirements can be as low as one milisecond. Moreover, when devices communicate with each other in addition to the human interaction, latency requirements goes lower since the devices can process the data faster [2]. Due to the increased data rate and decreased latency requirements, next generation mobile communications systems are expected to allow higher capacities and lower transmission delays. Therefore, 5G is foreseen to provide a "zero latency gigabit experience" and reduction of latency is considered as equally important as increased data rates. In fact, zero latency does not actually mean the absence of delay but it implies latency lower than 1 milisecond [2].

Since an important percentage of mobile traffic is a consequence of duplicated downloads of the same content, researchers are investigating efficacious caching strategies in order to prevent transmission of the same content over and over [3]. When the data is served from the caches located closer to the end user, the end-to-end transmission time of the content becomes smaller and smaller. To decrease the end-to-end latency in 5G cellular networks, we propose an intelligent caching and prefetching method using semantic inference technologies.

If the elements (e.g., base stations, edge clouds, gateways) of a mobile network were capable of inferring what the subsequent request will be, they would be able to prefetch that content before the actual request comes and store it in the cache allowing itself to serve it from the caches when requested. This would reduce the time passed during the retrieval of the data from the Internet when the actual request is raised and decrease the end-to-end latency. To enable the network elements reason on the requests, the meaning of each request must be known and understood by the computing entities. The metadata representing the meaning of the content have to be transmitted to the network elements. When the metadata arrives at the network elements, they will be able to make inferences using ontologies. In this work, we do not concentrate on the problem of generating the meaning of the content. Instead, we try to solve the problem of transmission of the metadata among hosts and the network elements.

The paper is organized as follows. In Section II, we present the three *W* questions of caching in cellular networks. We discuss the semantic caching and prefetching solution we are working on in Section III and then draw conclusions of this work-in-progress paper.

II. Challenges of Caching in Cellular Networks

In this section, we present the technical issues in caching in cellular networks and the related work.

A. 3W of Caching

Studies related to caching in cellular networks revealed that there are three main questions to answer when developing a caching method. These questions are 1) where to cache, 2) what to cache, and 3) what to release [3]. There are different answers for them in the literature but those answers all have some benefits and some drawbacks.

For the deployment position of the caches, two main options exist in 4G/5G networks. The first option is locating caches in the evolved packet core (EPC) as shown in Fig. 1. Deploying the caches at the EPC will increase the hit ratio of the caches since the cache will serve a huge number of users and

most of the traffic will flow through it. The deployment and maintainability of the caches will be easier when they are located at the EPCs.

The alternative option is deploying the caches at the radio access network (RAN). The main components of RAN are the eNodeBs and caches are attached to them as shown in Fig. 2. In typical cellular networks, there will be many eNodeBs deployed. Therefore, the storage capacity of the caches in each eNodeB will be smaller due to the financial issues. The number of users served per eNodeB is lower when compared to the EPC. For these reasons, the hit ratio in conventional RAN caching methods where prefetching is not employed, is smaller than the methods where the caches are located in the EPC. However, the latency when the data is served from the caches in the RAN will be smaller because of the proximity to the end users. Another advantage of caching at RAN is that backhaul links between EPC and RAN are not used when the data is served from the caches at the RAN. This helps to solve an important problem in the modern cellular networks which is bandwidth shortage in backhaul links by serving the data from RAN without using those links. When the data is served from the caches at EPC, links above the P-GW is not used but backhaul links between EPC and RAN are still used [3]. In Table I, we briefly summarize the advantages and disadvantage of these two options. Not to mention, the third option would be the hybrid approach that we set out of the scope of this paper.

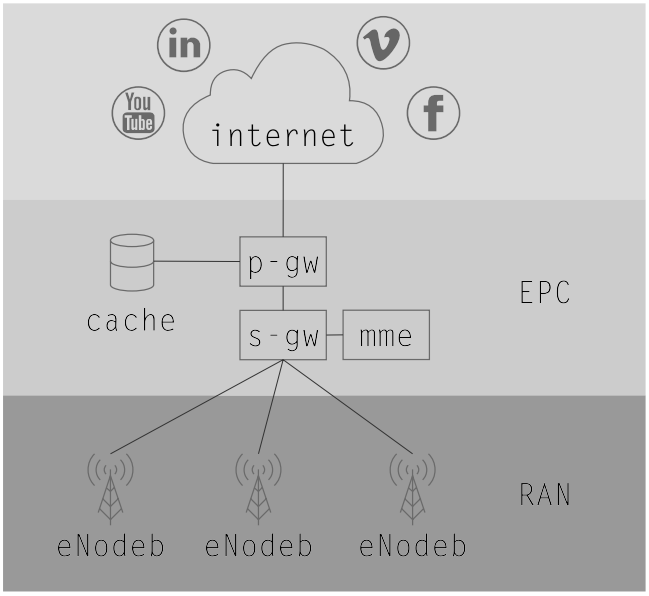

Fig. 1: A mobile cellular network architecture where caches are located at the EPC.

Caching helps to improve the effective bandwidth and decrease the end-to-end latency in the cellular networks. However, caching everything is not possible. The hardware cost for the storage is decreasing but contents generated by the end users are dramatically increasing. Therefore, the decision about what to cache should be made carefully. For example,

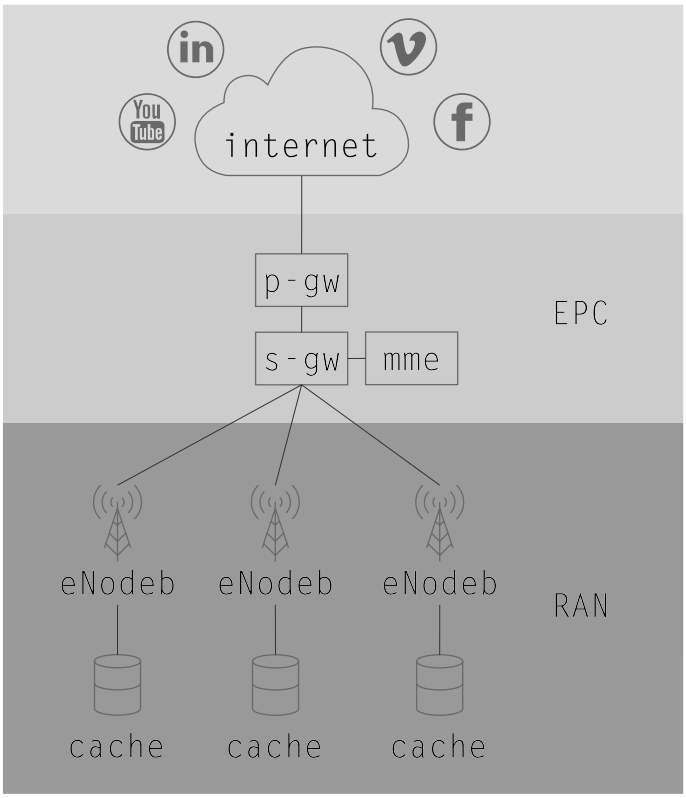

Fig. 2: A mobile cellular network architecture where caches are located at the RAN.

TABLE I: Comparison of caching at RAN and at EPC.

	EPC	**RAN**
Hit Ratio	Higher	Lower
Ease of Maintenance	Higher	Lower
Relaxing Backhaul Links	No	Yes
Decrease in Latency	Lower	Higher

the same content should not be stored at the neighboring eNodeBs because they can exchange data to be served to the end user. In order to increase the hit ratio of the caches, storing popular contents is the most common method whereas predicting future request is a challenging one.

Another important issue is deciding which contents should be removed from the cache because the cache will eventually be fully occupied. Long-established methods like first-in first-out (FIFO), least frequently used (LFU), and least recently used (LRU) may not be sufficient. Alternative methods like understanding that a cached content has lost its popularity may produce better solutions.

B. Existing Work

As mentioned above, there are some technical issues that are faced during all caching method implementations. A good amount of research has been done about these issues and they will be presented in this section.

A commonly used caching method is web caching. It takes the advantage of mobile traffic using hyper text transfer protocol (HTTP) for 82% of the time [4]. Web caching identifies content using their uniform resource locators (URL), which makes web caching a content aware caching method. When a web content is requested, it first checks whether a content with the same URL exists in the cache or not. If the content is in the cache, it passes that content to the user. To manage this, access frequency of each content and a table for URL of each content is stored. Web caching can be deployed on both RAN and EPC. Even if web caching is a commonly deployed method and it helps to prevent duplicate content transmission and decreases end-to-end latency, it has some disadvantages. First of all, it cannot avoid duplicate transmission of the same content when they have different URLs. For example, it cannot match two identical videos when they are located in different servers, which have different URLs. Also, temporary content (i.e. one-time used content) cannot be cached with this method. Finally, content updates are not observed with standard web caching method [5].

In [6], a proactive caching method called Proactive User Preference Profile (P-UPP) is proposed. It prefetches videos that are most likely to be requested by the users in a cell by calculating User Preference Profile of users in each cell. The reason for making predictions for each cell is that caches are located in eNodeBs in the cellular networks. P-UPP has the advantage of being capable of serving content from the caches even at the first request. When the caches are located at the RAN, the end-to-end latency will be very small because mobile devices and eNodeBs are very close to each other. Because of the fact that cache capacities must be small when they are located at the RAN, success rate of the predictions is very important in order not to fill the caches with unrelated data. In P-UPP, upon user arrivals or departures in a cell, video request probabilities are recalculated. If expected improvement of cache hit ratio is greater than a threshold, necessary video contents are preloaded to the caches. There is no pre-fetching at user events other than location changes. After a user makes a request, this method does not pre-fetch any content using the information behind that request. It does not benefit from the fact that a request coming from a user may signal another request that will come in a small amount of time.

III. Semantic Caching

As stated in the previous section, there are existing solutions to decrease latency in mobile networks using caching methods. Most of them cache the content after the first time it is served to the user. They locate caches in different places and they have different algorithms for perceiving duplicate data and storing content in the caches. However, they are not able to serve any content from the caches if the content has not been requested before. Only proactive methods like P-UPP are capable of serving data from the caches when the first time it is requested by prefetching the data before the content is requested. P-UPP predicts user events based on user preference profiles and does not respond to user events actively. On the other hand, our method brings the ability to prefetch data from Internet each time the user makes a request. It makes inferences on each request and predicts what will next requests of the user be. Depending on these predictions, necessary content is prefetched from Internet and stored in the caches. When the user requests a content that has been cached before thanks to previous inferences, that content is served directly from the caches without retrieving it from Internet.

A. Caching Strategy

Our caching strategy is composed of two main parts. One part is carried out in the user equipment and the other one is done on the mobile edge cloud. For each request of the user, metadata about the requested content should be generated on users mobile device. Labeling the devices, users and the content with machine-readable metadata improves the awareness of the devices. The network elements may use the metadata of the contents to name them and to derive useful information through reasoning. This metadata contains the information about the content.

The metadata language is determined as Web Ontology Language (OWL) due to its expressiveness and inference capability. The distribution of the metadata is pushed to the network layer instead of the higher layers, since we want to have an application and device independent architecture. The extension headers, *Hop-by-Hop Options* and *Destination Options* of IPv6 protocol are used as the carrier. The semantic definition of the device and the content is injected into the extension headers. If the application requires special handling at each hop on the route to destination, the metadata is located inside the Hop-by-Hop Options header since it is processed by the network elements such as the edge cloud. If the metadata is only required by the destination host, Destination Options header is used as the carrier. The metadata reaches to the network core along with the IP payload. On the network side, metadata is retrieved from the IP packet. If a content with the same metadata exists in the caches, it is served to the user without fetching it from Internet. Otherwise, content is fetched from Internet and served to user and stored in the caches with its metadata. Moreover, using metadata of the content, similar contents on the Internet are found they are fetched from Internet and stored in the caches as well, which makes this method a smart and proactive caching method that is able to respond user events spontaneously.

B. Location of The Caches

The decision of where to put the caches should be made according to the caching strategy. The reason is that, both places for deployment of the caches, namely EPC and RAN, has both advantages and disadvantages as explained in the previous section. Neither one is better than the other in all cases. If the proper selection is made depending on the caching method used, maximum benefit can be ensured.

Our caching method aims to be prepared for next requests of the users by predicting next user requests from the current request. Each request of a user has an implicit information

about what next requests that user might make. For example, when you make a request to search something from a search engine, the implicit information is that you are probably going to make a request to examine one or more of the search results soon. In our method, that implicit information is used to prefetch and cache contents that has a potential of being requested by the same user. Therefore, the main purpose is not serving a cached content to many users after its first usage but serving a cached content to a specific user at the first time it is demanded. Therefore, deployed caches do not need to be located at a place that is at the center of the mobile network. Hence, EPC loses the advantage of being at the center of the network and controlling too much traffic. On the other hand, eNodeBs serve to smaller number of users compared to the core network which becomes an advantage in our case. Location of the caches is an important factor for determining the time gained when the content is served from the caches. As caches get closer to the mobile devices, end-to-end latency is decreased and quality of service is improved. Assuming that all the other conditions are equal, placing caches in the RAN will have an absolute decrease, compared to caching within EPC, in the retrieval time of the content because of removal of multiple links of the end-to-end path to the content. Therefore, placing caches at RAN has a great advantage when the main purpose is decreasing the end-to-end latency.

C. Example Scenario

An example scenario of semantic caching is shown in Fig. 3. A user queries "Funny Videos" using a search engine. It is converted into an HTTP request and metadata describing the request is generated and inserted in the extension header of the IPv6 datagram carrying the HTTP request (step 1 in Fig. 3). That request is transferred to the eNodeB to which the user is currently connected (step 2). In the mobile edge cloud, the metadata is extracted from the extension header and searched in the cache (step 3). We assume the content is not found in the cache. Then, the HTTP request is sent to the EPC and the response is retrieved from the Internet (step 4.1). While the HTTP result is fetched from Internet, similar videos are found making inferences using the metadata at the mobile edge cloud. After the inference on potential subsequent requests, those inferred videos are prefetched from the Internet and stored in the cache (step 4.2). Steps 4.1 and 4.2 are carried out in parallel. After user gets the result of "Funny Videos" request, it will make a request for a prefetched content with a very high probability (step 5). The content is served to the user directly from the cache that will significantly reduce the end-to-end latency for subsequent requests (step 7). Moreover, if another user in the same cell wants to watch one of the videos that is in the first page of "Funny Videos" request, that user will retrieve it from the cache as well that will reduce the overall delay in the system.

IV. Conclusion

Because of the increase in the usage and capabilities of the mobile devices, the need for higher data rates and lesser end-to-end delay has emerged. In order to provide less than 1 milisecond delay in 5G mobile networks, the semantic caching method we propose in this work-in-progress paper can be employed. The semantic cache is able to serve content even at the first request of the content because of its proactive nature. It can respond to user events and populate caches accordingly, which is its main difference from the existing caching methods. As a future work, we plan the validate the semantic caching method using the NS3 simulator and quantify the advantage and disadvantages of this proposal. Furthermore, we will investigate the hybrid caching approach in a hierarchical way.

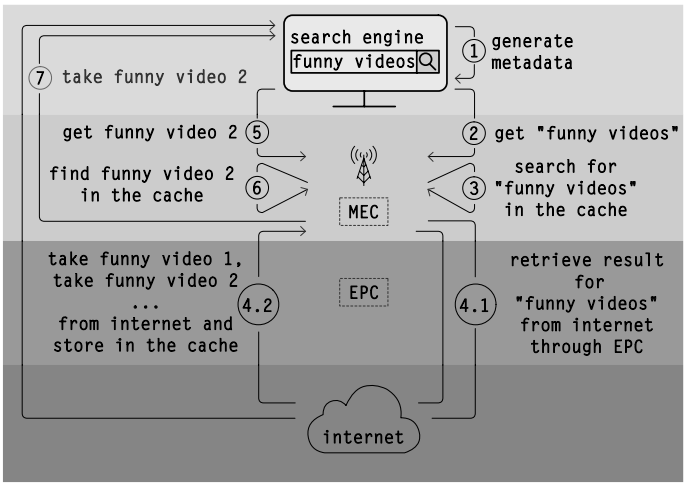

Fig. 3: An example scenario depicting the semantic caching using mobile edge computing (MEC).

V. Acknowledgment

This work is partially supported by TÜBİTAK under the grant number 115C064.

References

[1] 3GPP, "Service aspects; services and service capabilities," *Technical Specification*, 2010.

[2] Nokia Solutions and Networks, "5G use cases and requirements," *White Paper*, 2014.

[3] X. Wang, T. Taleb, A. Ksentini, and V. C. M. Leung, "Cache in the air: exploiting content caching and delivery techniques for 5G systems," *IEEE Communications Magazine*, vol. 52, pp. 131–139, 2014.

[4] J. Erman, A. Gerber, M. T. Hajiaghay, D. Pei, S. Sen, and O. Spatscheck, "To cache or not to cache: The 3G case," *IEEE Internet Computing*, vol. 15, pp. 27–34, 2011.

[5] S. Woo, J. Lee, E. Jeong, S. Ihm, S. Park, and K. Park, "Comparison of caching strategies in modern cellular backhaul networks," *Proc. of the ACM MobiSys*, 2013.

[6] H. Ahlehagh and S. Dey, "Video caching in radio access network: impact on delay and capacity," in *Proc. of the IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2012, pp. 2276–2281.