Wireless Caching in RAN

Subjects: Computer Science, Software Engineering

Contributor: Wei Chen

Caching has attracted much attention recently because it holds the promise of scaling the service capability of radio access networks (RANs). To realize caching, the physical layer and higher layers have to function together, with the aid of prediction and memory units, which substantially broadens the concept of cross-layer design to a multi-unit collaboration methodology.

Keywords: caching ; radio access networks ; virtual real-time services ; AI-empowered networks ; cross-layer design ; pricing ; recommendation ; edge compression ; proactive computation

1. Introduction

Modern radio access networks are capable of achieving data rates of Gbps, while they may still fail to meet the predicted bandwidth requirements of future networks. A recent report from Cisco [1] forecasts that mobile data traffic will grow to 77.49 EB per month in 2022. In theory, a human brain may process up to 100T bits per second [2]. As a result, a huge gap may exist between the future bandwidth demand and provision in next generation radio access networks (RANs). Unfortunately, on-demand transmission that dominates current RAN architectures has almost achieved its performance limits revealed by Shannon in 1948, given extensive development of physical layer techniques in the past decades. On the other hand, the radio spectrum has been over-allocated, while the overall energy consumption is explosive. Since the potential of on-demand transmission has been fully exploited, it is time to conceive novel transmission architectures for sixth generation (6G) networks [3] so as to scale its service capability. The cache-empowered RAN is one of the potential solutions that hold the promise of scaling service capability [4].

Caching techniques were originally developed for computer systems in the 1960s. Web caching was conceived for the Internet due to the explosively increasing number of websites in the 2000s. In contrast to on-demand transmission, caching allows proactive content placement before being requested, which has motivated some novel infrastructures such as information-centric networks (ICNs) and content delivery networks (CDNs).

More recently, caching has been found to substantially benefit data transmissions over harsh wireless channels and meet growing demands with restrained radio resources in various ways [5][6][7][8].

Though considerable literature on the subject of wireless caching exists, there is a need to revisit it from a cross-layer perspective, as shown in **Figure 1**.

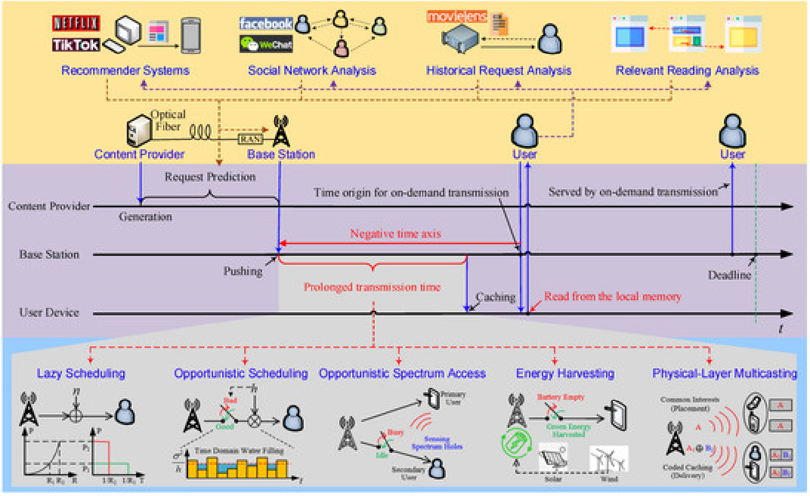

Figure 1. A unified framework for understanding caching gains from a time-domain perspective. Caching prolongs the transmission time, which enables various wireless techniques e.g., in **Table 1** that trade the transmission time for energy

and/or spectral efficiencies.

Table 1. Tradeoff Between the Transmission Time and SE/EE.

Transmission Techniques	Application Scenarios	How Is SE or EE Gain Attained?	Why Is Delay Increased?
Lazy Scheduling	Additive White Gaussian Noise Channels	Due to the convexity of Shannon capacity, EE is a decreasing function of the transmission power/rate.	Low data rate
Opportunistic Scheduling	Fading Channels	EE/SE is increased by time domain water-filling, or simply accessing good channels only.	Channel states remaining poor
Opportunistic Spectrum Access	Secondary Users	SE is increased by sensing and accessing idle timeslots or spectrum holes.	Spectrum remaining busy
Energy Harvesting	Renewable Energy Powered BSs/UEs	The renewable energy harvested from solar panels, wind turbines, or even the RF environment helps to save grid power.	No or little energy harvested
Physical-Layer Multicasting	Users with Common Interests	Multiple users located in the same cell are served by broadcasting a common signal to them.	Waiting for common requests

2. Proactive Service: Gains, Costs, and Needs

Without waiting for users' orders, a cache-empowered RAN provides proactive services.

2.1. Caching Gains: A Time-Domain Perspective

- Caching enables physical layer multicasting [9]. In theory, caching is capable of serving infinitely many users with a common request, thereby making RANs scalable. Classic on-demand transmission can seldom benefit from the multicasting gain because users seldom ask for a common message simultaneously. Aligning common requests in the time domain may, however, cause severe delay and damage Quality-of-Service (QoS). Proactive caching brings a solution to attain multicasting gain without inducing delay in data services. Even when users have different requests, judiciously designed coded caching strategies [10][11] allow RANs to enjoy the multicasting gain.

- Caching extends the tolerable transmission time, thereby bringing spectral efficiency (SE) or energy efficiency (EE) gains. Lazy scheduling [12], opportunistic scheduling [13][14], opportunistic spectrum access (OSA) [15], and energy harvesting (EH) [16] may increase the SE and EE. However, their applications are usually prohibited or limited due to their random transmission delay. Caching enables content transmission before user requests and hence substantially prolongs the delay tolerance.

- Caching enables low-complexity interference mitigation or alignment [17]. It is well known that a user can cancel a signal's interference based on prior knowledge about the message that the signal bears. Classic successive interference cancellation (SIC) decodes the interference first by treating the desired signal as noise. However, SIC can suffer from high complexity and error propagation. By contrast, caching provides reliable prior knowledge on the interfering signal, which significantly reduces the complexity of interference cancellation.

In many wireless techniques, there exists a fundamental tradeoff between transmission time and radio resource efficiency, as summarized in **Table 1**. In practice, however, trading the transmission time for energy efficiency (EE) or spectral efficiency (SE) is mostly prohibited due to the reactive service mode. When content items are transmitted upon user requests, an increased transmission time results in a large delay, leading to poor QoS.

Caching is a straightforward solution prolonging the transmission time by allowing RANs to push files to the network edge or a user's device before they are requested. In the reactive mode, a user's request time is usually regarded as the time origin of scheduling. Hence content placement is launched on the negative time-axis. In this way, we increase the transmission time while assuring that a user experiences small delay, as shown in **Figure 1**. In this case, the methods in **Table 1** can be exploited to increase the EE or SE. Though their data rate is low or unstable, caching allows a user to experience real-time services virtually.

Caching is expected to benefit the next generation RAN, e.g., 6G, in many aspects. A natural question to ask is which layer caching belongs to. If it only makes binary decisions on "to cache or not to cache" in each node, it is more like a network-layer protocol.

Instead of waiting for a user’s command, a RAN itself not only makes binary decisions on whether “to cache or not to cache”, but also determines “when and how to cache”. Thus, it requires novel functions beyond a bit-pipeline. In [12][13][14], threshold-based policies avoid pushing undemanded files that waste energy. When the demand probability for a content is below a certain threshold, its caching gain fails in compensating for the waste of radio resources. In addition, for popular files, caching too early results in the reduction of the available time for content pushing, resulting in the loss of SE/EE gains. Caching too late, however, may miss the request. To make caching practical, careful scheduling is desired in both placement and delivery phases.

2.2. Memory Cost to Be Paid for Caching

A cost to be paid arises from the memories at edge nodes and end devices, which are inexpensive but not free [4]. The memory cost is determined by not only how many bits are cached, but also how long they are cached [18][19].

The average memory size of user devices continues to grow. However, no matter how large a memory is, overflow occurs if the cached contents are never evicted. As a result, a content item should be evicted from the edge node if it becomes unpopular or outdated, or from a user’s device if it has been already read by this user. After its eviction, the occupied memory space will be released, which enables memory sharing in the time domain. The eviction time is a challenging issue. Evicting too early may cause possible future requests to be missed, evicting too late wastes the storage resource.

On the other hand, if a file is cached much earlier than its request time, the storage resource is also wasted. If it is cached too late, it may miss user requests, thereby losing the caching gain. In addition, it is inefficient to update memory when the channel remains poor or the spectrum is busy. Hence there is a need to jointly optimize pushing and memory updating, which generalizes the concept of cross-layer design as both wireless links and memories are involved. Such communication-storage coordination becomes very challenging with preference skewness, radio environment dynamics, and coded caching/prefetching.

3. Request Time Prediction: Beyond Content Popularity

Request time prediction is potentially highly beneficial in proactive caching. Unfortunately, conventional popularity based models, either static or time-varying, are content-specific. They mainly focus on the content popularity distribution among users.

3.1. Characterization of Random Request Time

Request time prediction relies on the fact, also observed in [4], that a content item is usually requested by a user at most once. We set a content item’s generation time to be the time origin. The item can be requested by a user at a random time after its generation, denoted by X, also referred to as the request delay. If it is never requested by the user, we regard the request delay to be $X=0^-$. Otherwise, the user will ask for it at $X\geq 0$. The accurate request delay X can hardly be predicted, but its probability density function (p.d.f.), denoted by p(x), is predictable. We shall refer to p(x) as the statistical request delay information (RDI), which characterizes our prediction about the request time [20].

RDI provides more knowledge than demand probability and popularity, because we can obtain a user’s demand probability α for a content item from its RDI, i.e., $\alpha=\int_0^\infty p(x)dx$. Further, if we assign lower indices i and k to indicate users and content items respectively, the popularity of item k can be characterized by $N_i\sum_k N_k$, in which $N_k=\sum_i \alpha_{ik}$ represents the expected total number of requests for item k.

3.2. RDI Estimation Methods

Artificial Intelligence (AI) and big data technologies provide powerful tools for understanding user behaviors in the time domain [21][22][23]. A time-varying popularity prediction for video clips can be found in [24][25], in which real data from YouTube and Facebook are used. In practice, the request time is also affected by one’s environments, activities, social connections, etc. For instance, one tends to watch video clips to kill time in the subway or during leisure time, but internet surfing is strictly prohibited while driving. Consequently, user-specific prediction brings together human behavior analysis, natural language processing (NLP), social networks, etc., leading to many cross-disciplinary research opportunities that include but are not limited to

- Learning a user’s historical requests and data rating [26][27],
- Exploiting the impact of social networks, recommendation systems, and search engines,
- Discovering relevant content using NLP,

- Analyzing a user behaviors, e.g., activities, mobilities, and localizations.

4. Fundamental Limits of Caching: A Cross-Layer Perspective

4.1. Communication Gains

Proactive caching prolongs the transmission time, which enables many possible energy- and/or spectral-efficient physical layer techniques. We are interested in how a content item is pushed given its RDI and what its EE/SE limit is. Quantitative case studies on the EE of pushing over additive white Gaussian noise (AWGN), multiple-input single-output (MISO), and fading channels are presented in [12][13][14], respectively. A user that tolerates a maximal delay of *T* seconds may request a content item having *B* bits. The AWGN channel has a normalized bandwidth and power spectral density of noise.

4.2. Memory Costs

As noted previously, a cost of caching is increased memory cost, which has to pay the memory cost, which can be reduced by efficiently reusing memory in the time domain. The memory cost is determined by not only how many bits are cached, but also how long they are cached. Memory is wasted if a content item is cached much earlier than being requested or evicted too late after being unpopular. Unfortunately, due to the lack of the request time prediction, how to reuse memory efficiently in the time domain has long been ignored.

Memory scheduling becomes more challenging in the following three scenarios. First, memory-efficient scheduling with coded caching remains open because the hit ratio of coded caching is still unknown. Second, the hit ratio can be increased by dropping less popular items when the memory is full. This makes the eviction policy more complicated [28]. Finally, the joint scheduling of memories and wireless links generalizes the concept of cross-layer design by involving both the communication and memory units. Deep learning and deep reinforcement learning are expected to play key roles in dealing with the dynamic nature of user requests and radio environments [29][30][31].

5. Pricing: Creating Incentive for Caching

5.1. Pricing Caching Service Using a Hierarchical Architecture

We conceive a hierarchical architecture with virtual network operators (VNOs) [32], as shown in **Figure 2**. A RAN sells its bandwidth to VNOs, which buy bandwidth to serve their associated users, either by on-demand transmission or caching. If a user cannot find the requested file from the local memory, her or his VNO has to buy bandwidth to serve it. A VNO charges its user for the data volume that the user requests, no matter how a requested file is served.

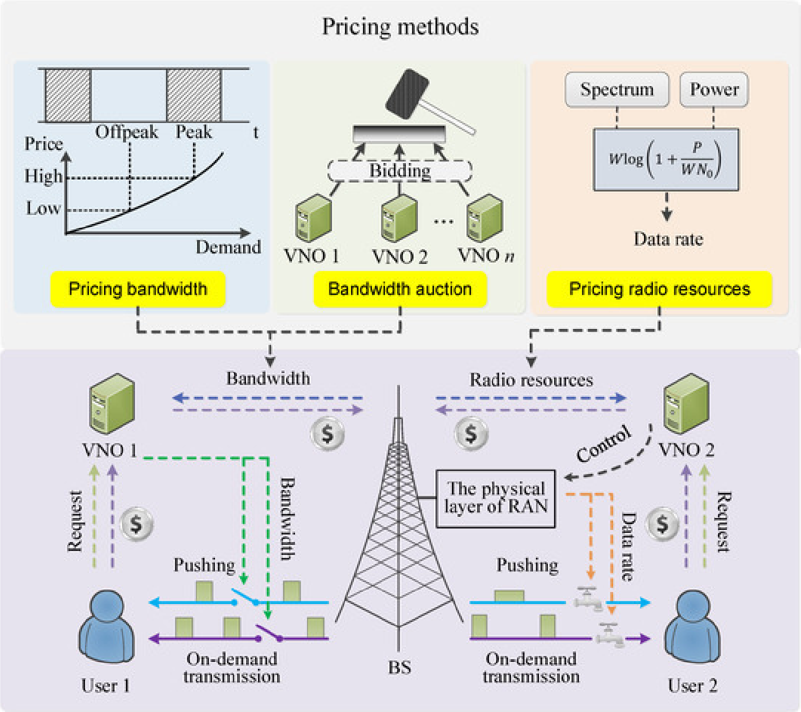

Figure 2. A hierarchical pricing infrastructure for cache-empowered RANs, in which various bandwidth or radio resource pricing mechanisms can be adopted.

A simple scenario in which VNOs schedule bandwidth only is discussed first. The RAN operator charges VNOs a higher bandwidth fee during peak times, because the price is determined by the demand-supply relationship from an economics perspective. If a user's requested file can be found in her or his local cache, the service cost is low. Otherwise, on-demand transmission has to be launched, even if the instantaneous cost is high. On the other hand, caching undesired content wastes the pushing cost. Consequently, VNOs have a strong incentive to maximize the cache hit ratio through accurate request prediction and careful scheduling. This incentive helps to better match the bandwidth demand and supply in the time domain [33]. An alternative way to advocate caching is nonlinear pricing in which the cost per unit spectrum increases with the total amount of spectrum acquired by a user.

Caching should reduce a user's cost for telecommunication services, while increasing the income of spectrum owners and/or RAN operators. The two goals seem to be contradictory, but can be achieved simultaneously due to caching gains. More specifically, the overall service costs are reduced due to the EE and SE gains of caching. Pushing popular items in off-peak time helps to reduce the bandwidth demand during peak times. As such, proactive caching better matches the bandwidth demand and supply in the time domain, which also broadens the cross-disciplinary research of economics and wireless networks.

A RAN may adopt an auction that allows VNOs to bid for bandwidth. In this case, the gap between bandwidth prices in peak and off-peak times can become even larger and hence caching saves more cost. Further, if a VNO fails in bidding bandwidth to serve its users, it fails in assuring the QoS, thereby losing users. Therefore, a bandwidth auction may not only increase the income of a RAN, but also eliminate VNOs with weak caching algorithms.

5.2. Pricing User Cooperation

Though user cooperation plays a central role in caching, selfish users may be unwilling to cooperate. Pricing is an effective tool to motivate user cooperation in various layers.

Caching-oriented pricing should reward users who contribute more memory for caching or private data for request prediction. A user's hit ratio is increased with more memory used for caching. However, more memory means higher device cost for a user. To reward users contributing more memory for caching, they should enjoy a discount on the telecommunication service fee. On the other hand, the accuracy of request prediction increases with more historical request data or more knowledge about social connections. Sharing these data means more risk in leaking a user's privacy, with which some users are seriously concerned. To gather more data for request prediction, a lower price should be charged for cooperative users.

5.3. Competition and Evolution

Multiple VNOs share the bandwidth provided by a common RAN. Such an infrastructure sharing model results in competition among VNOs, which may bid for bandwidth through auctions. If a VNO fails in getting bandwidth when it has to launch on-demand transmissions, its users will suffer from poor QoS. A VNO providing poor QoS frequently will lose its users. As a result, a VNO has to spend a lot of money to win the bandwidth auction if an on-demand transmission is necessary.

The greater the hit ratio is, the more profit the caching policy brings to a VNO [34]. With more profit, the VNO can afford a higher price to win the bandwidth auction when necessary, thereby assuring QoS. It may also reduce the service fee to attract cost-sensitive users. As a result, VNOs with low cache hit ratios will be either bankrupt due to high service cost or abandoned by users for poor QoS. In other words, the bandwidth auction not only brings more income for a RAN, but also motivates the evolution of prediction and caching policies.

5.4. Pricing Radio Resources, Memory, and Privacy

To fully unlock caching gains, VNOs should be allowed to control the physical layer directly. In particular, a RAN sells its radio resources to VNOs and lets a VNO decide how to use its bought power and spectrum, etc. In this case, VNOs have more freedom and incentive to optimize the SE or EE.

The memory cost should be considered in pricing. Intuitively, the hit ratio is increased if a user allocates more memory for caching, but more memory means a higher device cost paid by this user. Accordingly, users who are willing to contribute more memory to cache more data should be rewarded e.g., by offering them some discount, as noted previously.

Sharing the infrastructure, each VNO will announces its own pricing and reward policy. Each user then will chooses her or his favorite VNO based on the willingness of sharing private data, memory allocation, and her or his own preferences. In this context, mechanism design needs more quantitative study from a game-theoretic perspective.

6. Recommendation: Making RANs More Proactive

6.1. Joint Caching and Recommendation

Recommendation systems (RSs), long recognized as an area of computer science, have been widely used by content providers such as YouTube, Netflix, Tik-Tok, etc. News websites, and even search engines also “recommend” what may interest a user. Nowadays a large portion of data services are driven by RSs. Both RS and caching predict what a user is likely to be interested in, as noted in [4]. A RS aims to a user her or his favorite content items, while a cache-empowered RAN steps further by sending them to the user before being requested. Naturally, joint caching and recommendation (JCR) has attracted some recent attention.

Cache-friendly recommendation is a recent attempt in this area. Its intuitive idea is to push what an RS would recommend and let the user know. By this means, the hit ratio can be improved, as are caching gains. Meanwhile, the RS only recommends what it essentially wants to recommend and avoids showing a user what it is not interested in. In practice, however, the cached files may not be a user’s most favorite ones. In this case, cache-friendly recommendation was conceived to recommend content items that are cached but not the most favorable [35]. In addition, recommendation may enhance users’ common interests, thereby grouping them to achieve coded-caching or multicasting gain [36]. In both cases, a RAN can enjoy reduced peak-rate and improved SE and/or EE. A cost to be paid is that the user might be less satisfied with the RS if it frequently finds unwanted or useless recommendations. How undesired recommendations harm a user’s experience needs more experimental study.

6.2. After-Request Recommendation and Soft Hit

A more adventurous attempt is the recommendation after request, also referred to as the flexible recommendation. Specifically, when a user asks for a content item that has not been cached in the memory, the RS finds some relevant content items from the local cache, if there are any, and recommend them to the user [35]. If the user accepts the recommendation, a soft hit is achieved [37]. Otherwise, on-demand transmission will be launched to satisfy the user. It is a “win-win” solution for both users and RANs because the user enjoys low latency, better QoS, and price reduction by reading from local cache directly, while the RAN enjoys reduced peak-rate and improved SE/EE.

Though low latency sounds attractive, users sometimes need a stronger motivation to accept this “win-win” solution. A potential approach for boosting the soft hit ratio is to reduce or even waive the service fee of the recommended file. Though such a discount reduces a VNO’s income from serving the recommended file, it avoids the VNO spending much more money on bidding for peak-time bandwidth. After-request recommendation brings many cross-disciplinary research opportunities. For instance, how to discover relevant content from local cache needs investigation based on NLP or other cross-domain recommendation methods. In turn, the better QoS and lower price provided by caching improve a user’s willingness to accept recommendations, leading to an inherent interaction among caching, pricing, and recommendation that remains open.

References

1. Cisco. Cisco Visual Networking Index: Forecast and Trends, 2017–2022 White Paper, 2019. Available online: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.pdf (accessed on 17 May 2020).
2. Rappaport, T.S.; Xing, Y.; Kanhere, O.; Ju, S.; Madanayake, A.; Mandal, S.; Alkhateeb, A.; Trichopoulos, G.C. Wireless communications and applications above 100 GHz: Opportunities and challenges for 6G and beyond. IEEE Access 2019, 7, 78729–78757.
3. Letaief, K.B.; Chen, W.; Shi, Y.; Zhang, J.; Zhang, Y.-J.A. The roadmap to 6G–AI empowered wireless networks. IEEE Commun. Mag. 2019, 57, 84–90.
4. Paschos, G.; Baştuğ, E.; Land, I.; Caire, G.; Debbah, M. Wireless caching: Technical misconceptions and business barriers. IEEE Commun. Mag. 2016, 54, 16–22.
5. Paschos, G.S.; Iosifidis, G.; Tao, M.; Towsley, D.; Caire, G. The role of caching in future communication systems and networks. IEEE J. Sel. Areas Commun. 2018, 36, 1111–1125.

6. Bai, B.; Wang, L.; Han, Z.; Chen, W.; Svensson, T. Caching based socially-aware D2D communications in wireless content delivery networks: A hypergraph framework. IEEE Wirel. Commun. 2016, 23, 74–81.
7. Wang, L.; Wu, H.; Ding, Y.; Chen, W.; Poor, H.V. Hypergraph-based wireless distributed storage optimization for cellular D2D underlays. IEEE J. Sel. Areas Commun. 2016, 34, 2650–2666.
8. Yan, Q.; Chen, W.; Poor, H.V. Big data driven wireless communications: A human-in-the-loop pushing technique for 5G systems. IEEE Wirel. Commun. 2018, 25, 64–69.
9. Lu, Y.; Chen, W.; Poor, H.V. Multicast pushing with content request delay information. IEEE Trans. Commun. 2018, 66, 1078–1092.
10. Lu, Y.; Chen, W.; Poor, H.V. Coded joint pushing and caching with asynchronous user requests. IEEE J. Sel. Areas Commun. 2018, 36, 1843–1856.
11. Lu, Y.; Chen, W.; Poor, H.V. A unified framework for caching in arbitrary networks. In Proceedings of the 2018 IEEE 23rd International Conference on Digital Signal Processing (DSP), Shanghai, China, 19–21 November 2018; pp. 1–5.
12. Huang, W.; Chen, W.; Poor, H.V. Energy efficient pushing in AWGN channels based on content request delay information. IEEE Trans. Commun. 2018, 66, 3667–3682.
13. Lin, Z.; Chen, W. Content pushing over multiuser MISO downlinks with multicast beamforming and recommendation: A cross-layer approach. IEEE Trans. Commun. 2019, 67, 7263–7276.
14. Xie, Z.; Lin, Z.; Chen, W. Power and rate adaptive pushing over fading channels. IEEE Trans. Wirel. Commun. 2021. early access.
15. Li, C.; Chen, W. Content pushing over idle timeslots: Performance analysis and caching gains. IEEE Trans. Wirel. Commun. 2021. early access.
16. Zhou, S.; Gong, J.; Zhou, Z.; Chen, W.; Niu, Z. GreenDelivery: Proactive content caching and push with energy-harvesting-based small cells. IEEE Commun. Mag. 2015, 53, 142–149.
17. Maddah-Ali, M.A.; Niesen, U. Cache-aided interference channels. IEEE Trans. Inf. Theory 2019, 65, 1714–1724.
18. Chen, W.; Poor, H.V. Caching with time domain buffer sharing. IEEE Trans. Commun. 2019, 67, 2730–2745.
19. Xie, Z.; Chen, W. Storage-efficient edge caching with asynchronous user requests. IEEE Trans. Cogn. Commun. Netw. 2020, 6, 229–241.
20. Chen, W.; Poor, H.V. Content pushing with request delay information. IEEE Trans. Commun. 2017, 65, 1146–1161.
21. Bharath, B.N.; Nagananda, K.G.; Gündxuxz, D.; Poor, H.V. Caching with time-varying popularity profiles: A learning-theoretic perspective. IEEE Trans. Communn. 2018, 66, 3837–3847.
22. Lee, M.-C.; Molisch, A.F.; Sastry, N.; Raman, A. Individual preference probability modeling and parameterization for video content in wireless caching networks. IEEE/ACM Trans. Netw. 2019, 27, 676–690.
23. Yang, L.; Guo, X.; Wang, H.; Chen, W. A video popularity prediction scheme with attention-based LSTM and feature embedding. In Proceedings of the 2020 IEEE Global Communications Conference (GLOBECOM 2020), Taipei, Taiwan, 7–11 December 2020; pp. 1–6.
24. Yang, P.; Zhang, N.; Zhang, S.; Yu, L.; Zhang, J.; Shen, X. Content Popularity Prediction Towards Location-Aware Mobile Edge Caching. IEEE Trans. Multimed. 2019, 21, 915–929.
25. Tang, L.; Huang, Q.; Puntambekar, A.; Vigfusson, Y.; Lloyd, W.; Li, K. Popularity prediction of facebook videos for higher quality streaming. In Proceedings of the USENIX Annual Technical Conference (USENIX ATC), Santa Clara, CA, USA, 12–14 July 2017; pp. 111–123.
26. Wu, J.; Yang, C.; Chen, B. Proactive caching and bandwidth allocation in heterogeneous network by learning from historical number of requests. IEEE Trans. Commun. 2020, 68, 4394–4410.
27. Cheng, P.; Ma, C.; Ding, M.; Hu, Y.; Lin, Z.; Li, Y.; Vucetic, B. Localized small cell caching: A machine learning approach based on rating data. IEEE Trans. Commun. 2019, 67, 1663–1676.
28. Hui, H.; Chen, W.; Wang, L. Caching with finite buffer and request delay information: A markov decision process approach. IEEE Trans. Wirel. Commun. 2020, 19, 5148–5161.
29. Li, L.; Xu, Y.; Yin, J.; Liang, W.; Li, X.; Chen, W.; Han, Z. Deep reinforcement learning approaches for content caching in cache-enabled D2D networks. IEEE Internet Things J. 2020, 7, 544–557.
30. Li, L.; Cheng, Q.; Tang, X.; Bai, T.; Chen, W.; Ding, Z.; Han, Z. Resource allocation for NOMA-MEC systems in ultra-dense networks: A learning aided mean-field game approach. IEEE Trans. Wirel. Commun. 2021, 20, 1487–1500.

31. Chen, Q.; Wang, W.; Chen, W.; Yu, F.R.; Zhang, Z. Cache-enabled multicast content pushing with structured deep learning. IEEE J. Sel. Areas Commun. 2021, 39, 2135–2149.

32. Huang, W.; Chen, W.; Poor, H.V. Request delay-based pricing for proactive caching: A stackelberg game approach. IEEE Trans. Wirel. Commun. 2019, 18, 2903–2918.

33. Lin, Z.; Huang, W.; Chen, W. Bandwidth and storage efficient caching based on dynamic programming and reinforcement learning. IEEE Wirel. Commun. Lett. 2020, 9, 206–209.

34. Hui, H.; Chen, W. A pricing-based joint scheduling of pushing and on-demand transmission over shared spectrum. In Proceedings of the 2020 IEEE Global Communications Conference (GLOBECOM 2020), Taipei, Taiwan, 7–11 December 2020; pp. 1–5.

35. Chatzieleftheriou, L.E.; Karaliopoulos, M.; Koutsopoulos, I. Jointly optimizing content caching and recommendations in small cell networks. IEEE Trans. Mob. Comput. 2019, 18, 125–138.

36. Zhu, B.; Chen, W. Coded caching with moderate recommendation: Balancing delivery rate and quality of experience. IEEE Wirel. Commun. Lett. 2019, 8, 1456–1459.

37. Sermpezis, P.; Giannakas, T.; Spyropoulos, T.; Vigneri, L. Soft cache hits: Improving performance through recommendation and delivery of related content. IEEE J. Sel. Areas Commun. 2018, 36, 1300–1313.

Retrieved from https://encyclopedia.pub/entry/history/show/32478