

CONTENT EFFICIENT EMBEDDING MECHANISM AND ARTIFICIAL INTELLIGENCE-BASED ALGORITHM IN CDN CACHES

Khasanov N.N.

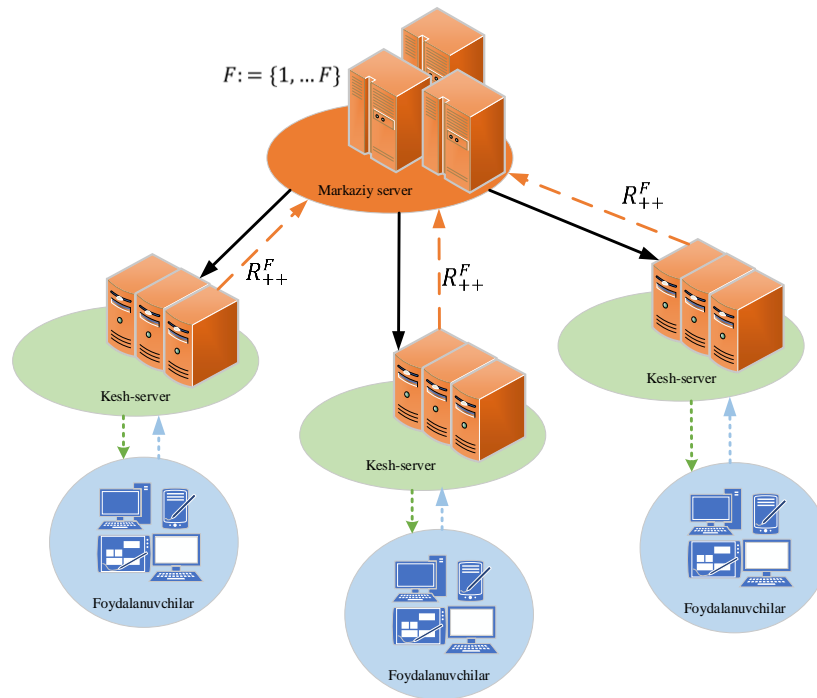
TATU named after Muhammad al-Khwarazmi, independent seeker

Keywords: CDN, cache server, algorithm, artificial intelligence, central server, memory, network, deployment, hit ratio.

In recent years, network traffic flowing through the internet is increasing, and this puts the task of efficiently distributing traffic in the appropriate way. In the implementation of the task, mainly cloud technologies and virtualization technologies are used, which increase the performance of the internet network. In addition to these technologies, the technology of more content delivery networks (CDN - Content Delivery Networks) was also produced to solve several problems of the internet network. This work examines the mechanism and artificial intelligence-based algorithm for efficiently locating content on CDN networks.

Tarmoq markaziy serveridan, $N:=\{1,\dots,N\}$ bo'lgan kesh-serverlardan va har bir kesh bilan bog'langan bir qancha foydalanuvchilardan iborat. Jarayonni soddalashtirish uchun biz har bir foydalanuvchi bitta kesh server bilan bog'langan holatni ko'rib chiqamiz. Bunda faqat bitta markaziy server hisobga olingan bo'lsada, mavjud formulani bir nechta markaziy serverlar mavjud tarmoq infratuzilmasiga osongina kengaytirish mumkin. Foydalanuvchilarning fayl so'rovlari kesh-serverlarda qabul qilinadi va fayl mavjudligiga qarab lokal taqdim etiladi yoki markaziy serverga yo'naltiriladi. Har bir kesh-server to'liq kontent kutubxonasining faqat bir qismini saqlaydi. Bundan maqsad kesh-serverlarda saqlanishi kerak bo'lgan fayllar to'plamini aniqlashdan iborat.

Kontent oqimini tahlil qilish uchun tarmoq tuzilishi ma'lumot paketlari tomonidan qabul qilingan marshrutlarga bog'liq bo'lgan graf sifatida modellashtirilgan. Graf topologiyasining turini tanlash CDN bilan mavjud keshlash tizimiga bog'liq.



1-rasm. Foydalanuvchilar tomonidan asosiy serverga murojaat topologiyasi

1-rasmda keltirilgan topologiya an'anaviy CDN o'rnatishda keng qo'llanilgan. Markaziy serverdan yuklab olingan har qanday fayl keshdan o'tganligi sababli, agar zarurat bo'lsa, u kesh-serverda saqlanishi mumkin. Shunday qilib, kontentni joylashtirish muammosi ma'lum bir fayl keshda saqlanishi yoki chiqarib tashlanishi haqida qaror qabul qilish muammosiga aylanadi. Bu ish uchun LRU va LFU kabi onlayn almashtirish algoritmlari keng o'rganilgan. Bundan farqli o'laroq, yuqoridagi topologiya mobil kichik sotalar orqali chegaraviy keshlashni ta'minlaydigan keyingi avlod sotali tizimlar uchun taklif qilingan. Eng yuqori yuklamali vaqt oralig'ida ko'pchilik kontentni kesh-serverning o'zidan yuklab olish talab qilinadi, shuning uchun past kechikish va yuqori xarajat foydalanuvchi tomonidan markaziy serverga ulanish faqat vaqti-vaqti bilan faollashadi.

Quyida markaziy server va N kesh-serverdan iborat umumiy CDN sozlamalari ko'rib chiqiladi. i – kesh-serverning saqlash hajmi M_i bilan belgilanadi va bitlarda o'lchanadi. Markaziy serverda $f \in F := \{1, \dots, F\}$ sifatida indekslangan fayllar kutubxonasi mavjud. f faylining o'lchami s_f bilan belgilanadi va u ham bitlarda o'lchanadi. Fayl o'lchamlari $s \in RF_{++}$ vektoriga yig'iladi. Vaqt slotlarga bo'linadi va vaqt sloti $t = 1, 2, \dots$ sifatida indekslanadi. T vaqtida i -keshda saqlangan fayllar to'plami $F_i(t) \in F$ bilan belgilanadi. Shu bilan bir qatorda, i -kesh uchun $m_i(t)$ saqlash vektori $f \in F_i(t)$ uchun $m_{if}(t) = 1$, aks holda nol sifatida aniqlanadi. Agar saqlash vektori $m_i(t)$, $sTm_i(t) \leq M_i$ ni qanoatlantirsa, i -kesh uchun yaroqli deb hisoblanadi. U_i i -keshga so'rov yuboradigan foydalanuvchilar

to'plami bo'lsin. Turli serverlarda tarmoq sig'imi va xizmat ko'rsatish nisbati cheklangan bo'lishi mumkinligi sababli, U_i -dagi foydalanuvchilar i -keshdan har bir vaqt slotida maksimal umumiy tezlikda C_c i bit va C_r i bitni markaziy serverdan har bir vaqt slotida olishlari mumkin deb hisoblanadi. Har bir t vaqt oralig'ida f fayli uchun keshdagi i jami talab $dif(t)$ bilan belgilanadi, u bitlarda o'lchanadi va $di(t) \in RF +$ vektoriga yig'iladi. Berilgan talab $dif(t)$ ni hisobga olgan holda, agar $mif(t) = 1$ bo'lsa, xizmat siyosati lokal sifatida xizmat ko'rsatish va aks holda so'rovni qayta yo'naltirishni amalga oshiradi. Barcha talablar keyinchalik qondirilgan deb faraz qilsak, i -kesh serverdan umumiy oqim tezligi $mT_i(t)d_i(t)$ tomonidan, markaziy serverdan U_i -da foydalanuvchilarga oqim tezligi esa $1T d_i(t) - mT_i(t)d_i(t)$ ifoda tomonidan beriladi, har ikkala vaqt slotida bitlar bilan o'lchanadi.

Agar foydalanuvchi talablari oldindan ma'lum bo'lsa, optimal kontentni joylashtirish $m_i(t)$ da butun sonli dasturlash muammosini hal qilishni o'z ichiga oladi[1]. G'oya, tegishli xarajatlarni tarmoq oqimlari bilan bog'lash va umumiy xarajatlarni kamaytiradigan faylni joylashtirishni topishdir. Hozirgi kontekstda i -kesh serverga ulangan so'nggi mil ulanishlaridagi oqimlar bilan bog'liq bo'lgan xarajat funksiyalarini $\chi_i: R \rightarrow R$ va i -kesh server bilan bog'langan foydalanuvchi bilan markaziy server o'rtasidagi oqimlarga mos keladigan $\varphi_i: R \rightarrow R$ ni aniqlash kerak (1-rasmdagi ko'k rangli nuqtali chiziqlar). Boshqa tomondan, 1-rasmda keltirilgan topologiya uchun $\varphi_i()$ i -keshga ulangan teskari ulanishlar va oxirgi mil ulanishlaridagi oqimlarni jazolaydigan jamlangan xarajatlar funksiyasini ifodalaydi.

Shunday qilib, har bir t vaqtida kontentni joylashtirish muammosi quyidagi butun sonli dastur sifatida shakllantirilishi mumkin:

$$\min_{\{m_i(t) \in [0,1]^F\}_i} \sum_{i,f} \chi_i(m_{if}(t)d_{if}(t)) + \sum_{i,f} \varphi_i(d_{if}(t)(1-m_{if}(t))) \quad (1)$$

$$s.t. \mathbf{m}_i^T(t)d_i(t) \leq C_i^c, \quad 1^T d_i(t) - \mathbf{m}_i^T(t)d_i(t) \leq C_i^r \quad \forall i \in N \quad (2)$$

$$s^T m_i(t) \leq M_i \quad \forall i \in N \quad (3)$$

Ya'ni, maqsad har bir t vaqtida minimal xarajatlarga olib keladigan $\{m_i(t)\}$ joylashtirish vektorlarini topishdir (1). (2) cheklov jamlangan oqimlarning tegishli ulanish sig'imlaridan oshmasligini ta'minlaydi, 3-ifoda esa ko'pi bilan M_i bitlarning i -chi keshda saqlanishini ta'minlaydi.

Foydalanuvchilarning fayllarni joylashtirish hajmiga bo'lgan talablari $d_t(t)$ oldindan aniq bo'la olmasligini hisobga oladigan bo'lsak, ushbu ifodaning yechilishi

imkonsiz bo'radi. Ushbu masalani yechishda ikkita soddalashtirish tenglamalari taklif etiladi.

Har bir i va t uchun $x_{if}(t) = m_{if}(t)d_{if}(t)$ va $y_{if}(t) = (1 - m_{if})d_{if}(t)$ ifoda mos ravishda kesh va markaziy server orqali U_i foydalanuvchilari uchun kutilayotgan oqimlarni bildiruvchi o'zgaruvchilar bo'lsin. O'zgaruvchilarning ikkita to'plami mos ravishda $N \times F$ haqiqiy $X(t)$ va $Y(t)$ matritsalariga yig'iladi. $m_{if}(t)$ dagi butun son cheklovi $x_{if}(t)$ va $y_{if}(t)$ ga manfiy bo'lmagan haqiqiy qiymatlarni olish imkonini berish orqali soddalashtiriladi (4). Bundan tashqari, kutilayotgan oqimlar T vaqt oralig'ida o'rtacha talabni qondiradi, ya'ni

$$\frac{1}{T} \sum_{t=1}^T (x_{if}(t) + y_{if}(t) - d_{if}(t)) = 0 \quad (4)$$

Bunday soddalashtirish zarur, chunki amaliy tizimlar uchun dastlabki $x_{if}(t) + y_{if}(t) = d_{if}(t)$ munosabatlarini o'rnatib bo'lmaydi. Buning o'rniga kelajakdagi talablarni hisobga olgan holda oqimlar oldindan baholanadi. Kutilayotgan oqimlar keyinchalik kontentni joylashtirish bo'yicha qaror qabul qilish uchun ishlatiladi. E'tibor bering, kontentni yetkazib berish vaqtidagi haqiqiy oqimlar har xil bo'lishi mumkin va foydalanuvchining aniq talablariga javob berishi kerak. Biroq, kutilayotgan oqimlarning kiritilishi bizga joylashtirish va kontentning ommabopligini kuzatish imkonini beruvchi mexanizm bo'lib xizmat qiladi. Bunday cheklovlar (over-the-horizon) odatda tarmoq holatlari ketma-ket ochiladigan onlayn resurslarni taqsimlash sozlamalarida qo'llaniladi[2]. Ushbu ikkita soddalashtirish bizga $m_i(t)$ saqlash o'zgaruvchisini yo'q qilishga imkon beradi va T vaqt oralig'ida kutilayotgan $x_{if}(t)$ va $y_{if}(t)$ oqimlari nuqtai nazaridan muammo tug'diradi,

$$\min_{X(t), Y(t) \in \mathbf{R}_+^{N \times F}} \frac{1}{T} \sum_{t=1}^T \sum_{i,f} (x_i(x_{if}(t)) + \varphi_i(y_{if}(t))) \quad (5)$$

$$s.t. \sum_f x_{if}(t) \leq C_i^c \quad \sum_f y_{if}(t) \leq C_i^r \quad \forall i \in N \quad (6)$$

$$\frac{1}{T} \sum_{t=1}^T (x_{if}(t) + y_{if}(t) - d_{if}(t)) = 0 \quad \forall i \in N, \forall f \in F \quad (7)$$

Agar talablar oldindan ma'lum bo'lsa, (5-7) muammoni oflayn rejimda hal qilish mumkin. Biroq, hozirgi holatda, amallarni ketma-ket bajarishga imkon beruvchi onlayn algoritmi taqdim etish va asimptotik jihatdan optimalga yaqin ekanligini ko'rsatish zarur. Algoritmning umumiy g'oyasini quyidagicha umumlashtirib tushintirish mumkin:

- t -chi vaqt oralig'ining boshida va joriy talablar $D(t)$ aniqlanmaguncha kutilayotgan $X(t)$ va $Y(t)$ oqimlarini aniqlash;
- (3) saqlash cheklovlariga rioya qilgan holda $\{X(t), Y(t)\}$ munosabat asosida $m_i(t)$ to'g'ri joylashtirish qarorlarini qabul qilish;
- fayl keshda joylashgan bo'lsa foydalanuvchi talablariga odatdagidek lokal sifatida, aks holda, markaziy server orqali xizmat qilish.

Kutilayotgan oqimlarni muddatidan oldin beradigan birinchi qadam eng muhimi bo'lib, har bir faylning ommabopligini to'g'ri kuzatish va tarmoq sig'imi cheklovlariga rioya qilish uchun javobgar bo'ladi (2). Ikkinchi qadam zaruriy qadam, chunki kutilayotgan oqimlar saqlash cheklovlariga rioya qilish uchun mo'ljallanmagan, bu aniq belgilanishi kerak. Uchinchi bosqichda CDN qarorlari qabul qilingan talab vektoriga bog'liq. Agar tarmoq yetarli darajada ta'minlangan bo'lsa, (2) sig'im cheklovlari har doim birinchi bosqichdanoq qondiriladi.

A0. Tahlil qilishni soddalashtirish va mazmunli natijalarga erishish uchun ushbu bo'limda muammo parametrlari bo'yicha ma'lum muntazamlik farazlari kiritiladi. Bu yerda talab qilinadigan farazlar kuchliroq bo'lsada, ular aniq standart hisoblanadi va talablar hamda xarajatlarning ba'zi patologik tanlovlarini bartaraf etish uchun talab qilinadi.

A1. Tarmoq sig'imi umumiy talablarni qondirish uchun yetarlicha katta, ya'ni

$$\sum_f d_{if}(t) \leq C_i^c + C_i^r \quad (8)$$

A2. $\chi_i(u)$ va $\varphi_i(v)$ jarima funksiyalari $0 \leq x_2 \leq x_1 \leq C_i^c$ va $0 \leq y_2 \leq y_1 \leq C_i^r$ oraliqda quyidagicha aniqlanadi:

$$\begin{aligned} m_x(x_1 - x_2) &\leq \chi'_i(x_1) - \chi'_i(x_2) \leq L_x(x_1 - x_2) \\ m_\varphi(y_1 - y_2) &\leq \varphi'_i(y_1) - \varphi'_i(y_2) \leq L_\varphi(y_1 - y_2) \end{aligned} \quad (9)$$

bu yerda: m_x , m_φ , L_x , va L_φ parametrlarning hammasi musbat.

Ulardan, agar tarmoq ortig'i bilan ta'minlangan bo'lsa, (A1) faraz amal qiladi. Boshqa tomondan, (A2) faraz xarajat funksiyalarini to'g'ri tanlash orqali qondirilishi mumkin. Qavariq optimallashtirish algoritmlari kontekstida kuchli qavariqlik va Lipschitz talablari standart sanaladi [3] hamda jarima funksiyalarida kvadratik yuqori va pastki chegaralarni nazarda tutadi. Ta'kidlash joizki, avvalroq kiritilgan kvadratik jarima funksiyasi (A2) talabga javob beradi. Shuningdek, tartibga solish muddati bilan to'ldirilgan chiziqli xarajatlar funksiyasidan foydalanish mumkin. Shunga o'xshash, xarajat funksiyasi $\chi_i(x) = kx_2(C_i - x) + k_i$, birinchi muddat xizmat ko'rsatish darajasi C_i bo'lgan M/M/1 navbatidagi

pakatlarning o'rtacha sonini kamaytiradi, shuningdek, $x_{max} < C_i$ bilan $[0, x_{max}]$ oralig'ida (A2) talablarini qondiradi.

Ushbu bo'limdagi asimptotik natijalar, shuningdek, $\{\lambda_{if}(0)\}$ boshlang'ich ikkilamchi iteratsiyalarni quyidagi ikkita shartga rioya qiladigan tarzda tanlashni talab qiladi:

$$\begin{aligned} \lambda_{if}(0) &\geq \min\{x'_i(0), \varphi'_i(0)\}; \forall i, f \\ \sum_{f=1}^F \lambda_{if}(0); \forall i \end{aligned} \quad (10)$$

Bu yerda $\Delta := \max\{L_\chi, L_\varphi\}(C_i^c + C_i^r) + F \max\{\chi'_i(0), \varphi'_i(0)\}$. Navbatdagi muhokama uchun $L := \{L_\chi, L_\varphi\}$ va $m := \min\{m_\chi, m_\varphi\}$ bo'lsin. Yuqorida keltirilgan ifodalarning asosiy natijasini ko'rsatishdan oldin, birinchi navbatda $\{\lambda_{if}(t)\}$ ning chegaralanganligiga oid quyidagi oraliq lemma o'rnatiladi. 2-Lemma. (A1) - (A2) va $0 < \mu < m$ uchun, (10) ifodaga muvofiq barcha $t \geq 1$ uchun quyidagi chegaralarga amal qiladi,

$$\begin{aligned} \lambda_{if}(t) &\geq \min\{x'_i(0), \varphi'_i(0)\} \\ \forall i \in N, f \in F \\ \sum_{f=1}^F \lambda_{if}(t) &\leq \Delta \\ \forall i \in N \end{aligned} \quad (11)$$

2-Lemma, agar to'g'ri ishga tushirilsa va μ uchun yetarlicha kichik bo'lsa, ikkilamchi iteratsiya barcha $t \geq 1$ uchun ma'lum chegaralar ichida qolishini ta'minlaydi (11). (A1) chegarasi qondirilsa, natija $\{d_{if}(t)\}$ talablaridan qat'iy nazar o'rinli bo'ladi.

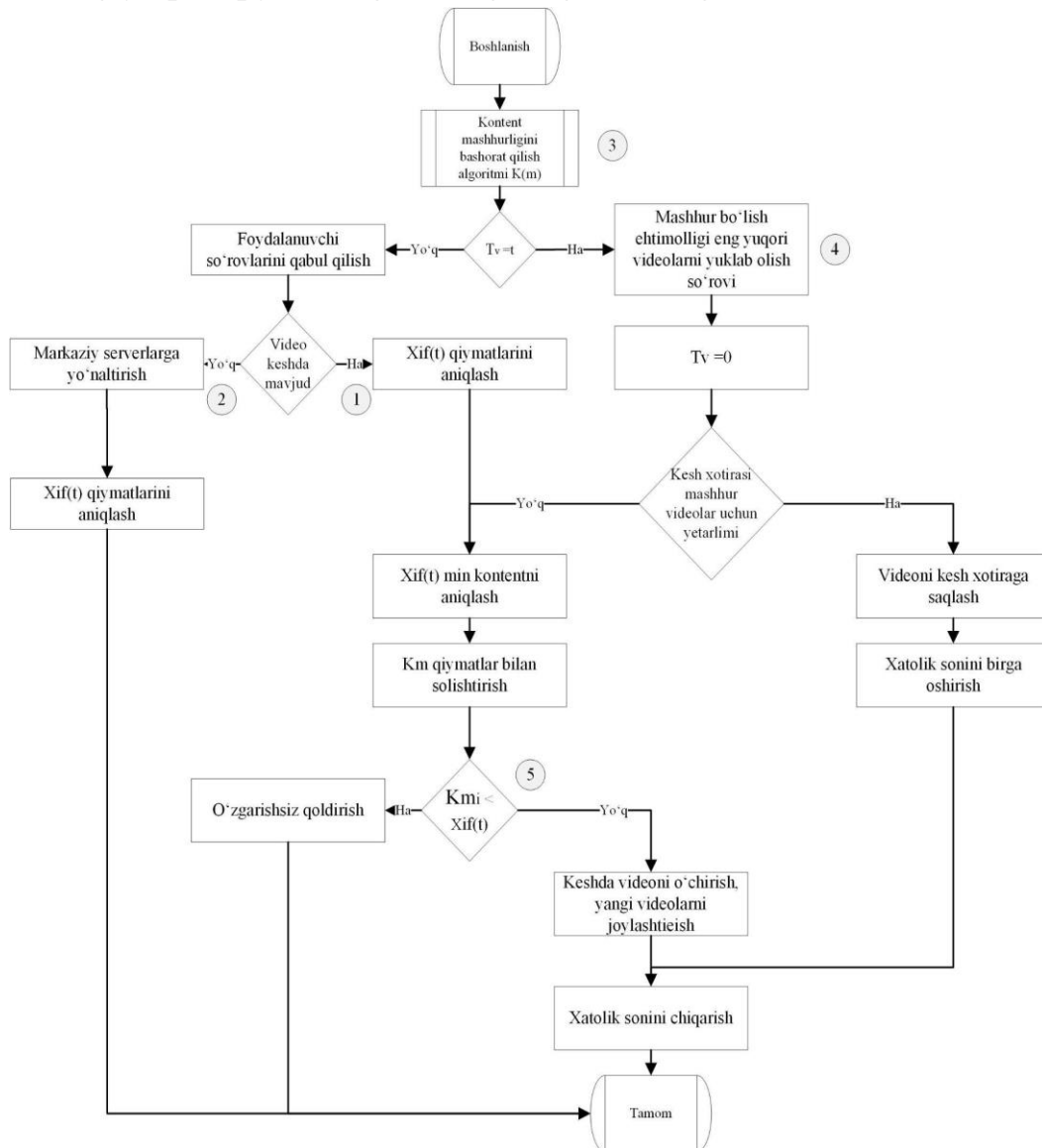
Yakuniy natijaga erishish uchun $\alpha_i(t)$ va $\beta_i(t)$ Lagranj ko'paytmalarini mos ravishda (6) dagi cheklovlar bilan bog'lash kerak. Xuddi shunday, $\xi_{if}(t)$ va $\zeta_{if}(t)$ cheklovlarni (5) mos ravishda $x_{if}(t) \geq 0$ va $y_{if}(t) \geq 0$ manfiy bo'lmagan cheklovlar bilan bog'lash kerak. Nihoyat, Lagranj ko'paytmasini v_{if} bilan (7) bog'lab, to'liq Lagranjni quyidagicha yozish mumkin

$$\begin{aligned} L_T \left(\{X(t), Y(t), \alpha(t), \beta(t), \xi(t), \zeta(t)\}_{t=1}^T, \nu \right) = \\ \frac{1}{T} \sum_{t=1}^T \sum_{i,f} \left(x_i(x_{if}(t)) + (\alpha_i(t) - \xi_{if}(t) - \nu_{if})x_{if}(t) + \varphi_i(y_{if}(t)) + (\beta_i(t) - \zeta_{if}(t) - \nu_{if})y_{if}(t) - C_i^c \alpha_i(t) - C_i^r \beta_i(t) + \nu_{if} d_{if}(t) \right) \end{aligned} \quad (12)$$

bu yerda: ajratib ko'rsatilgan miqdorlar barcha i va f uchun tegishli yig'ilgan o'zgaruvchilarni bildiradi. $Z(t)$ quyidagi o'zgaruvchilarni jamlagan deb olsak:

$$\{X(t), Y(t), \alpha(t), \beta(t), \xi(t), \zeta(t)\}_{t=1}^T \quad (13)$$

Talablarning vaqtinchalik o'zgarishiga nisbatan moslashuvchanlik ham amalda muhim ahamiyatga ega. Masalan, $\{d_{if}(t)\} \geq 1$ talablari odatda mustaqil va bir tekis taqsimlanmagan, balki vaqtinchalik yemirilish va korrelyatsiyani namoyish etadi. Xususan, vaqt o'tishi bilan kontentning ommabopligi ko'pincha kamayadi[4]. Masalan, [5]da talablar ko'pincha kunlik yondashuvga ega bo'lib, kunning ma'lum vaqtlarida eng yuqori qiymatlarga erishganligi kuzatilgan.



2-rasm. Sun'iy intellektga asoslangan kontentni optimal keshlash algoritmi

Algoritmning ishlashi bo'yicha qo'shimcha turtki olish uchun λ ni talabni qondirmaslik bilan bog'liq yashirin narxlar deb o'ylash maqsadga muvofiq. Kutilayotgan oqimlar tomonidan istalgan t vaqtida talab qondirilmasa yoki $x_{if}(t) + y_{if}(t) - d_{if}(t)$ manfiy bo'lganda narxlar oshadi. Yangilashning asosiy

bosqichlarida kutilayotgan oqimlar soya narxlarining yuqori yoki pastligiga qarab sozlanadi. Masalan, agar talablar turg'un bo'lsa, kutilayotgan oqimlar ularni shunday kuzatib boradiki, $x_{if}(t) + y_{if}(t) - d_{if}(t)$ farqi unchalik katta bo'lmaydi.

Oddiy misol sifatida foydalanuvchidan i talabi barcha t vaqt uchun nolga teng bo'lgan f faylini ko'rib chiqaylik. Bunday fayl uchun, agar $l_{if}(0) = 0$ bo'lsa, u barcha t vaqt uchun nol bo'lib qoladi. Darhaqiqat, initsializatsiya boshqacha bo'lsa ham, qo'sh o'zgaruvchilar nolga yaqinlashadi, chunki $x'_{if}(t)$ va $y'_{if}(t)$ har doim $l_{if}(0) > 0$ bo'lganda qat'iy musbat bo'ladi. Shuningdek, agar ma'lum bir $d_{if}(t)$ barcha t uchun yuqori bo'lib qolsa ham shunday bo'ladi deyish mumkin. Argumentni kengaytiradigan bo'lsak, $l_{if}(t)$ qo'sh o'zgaruvchisi $d_{if}(t)$ talablarini "kuzatadi", garchi uning kuzatish qobiliyati qadam o'lchami μ orqali cheklangan bo'lsa ham. Shunday qilib, $l_{if}(t)$ ning monoton funksiyalari bo'lgan kutilayotgan $\{x'_{if}(t), y'_{if}(t)\}$ oqimlari ham fayllarning ommabopligini kuzatib boradi. Bundan tashqari, talablarning vaqti-vaqti bilan ko'tarilishi yoki pasayishlari dual algoritmgaga xos bo'lgan o'rtacha effekt tufayli $l_{if}(t)$ ning evolyutsiyasiga minimal ta'sir ko'rsatadi.

Umumiy birinchi topologiya doirasida kontentni joylashtirish vaqti-vaqti bilan amalga oshirilishi kerak, chunki u bilan bog'liq kontentni olish xarajatlari yuqori. Tv up deb belgilangan yangilanish oralig'i bir necha soatdan kungacha o'zgarib turadi va keshdagi fayllar har bir bunday intervalning boshida yangilanadi. Buning uchun boshqa kontentni joylashtirish algoritmi ishlab chiqilgan bo'lib uning tushilishi 2-rasmda keltirilgan.

Ishlab chiqilgan algoritm: Birinchidan, yuqori $x_{if}(t)$ qiymatlari bo'lgan fayllar ommabop va keshlangan deb hisoblanadi. Ya'ni, har bir yangilanish bilan kesh saqlanadigan fayllar eng katta $x_{if}(t)$ qiymatlari bo'lgan fayllarga mos kelishini ta'minlaydi. Bunda fayllarning ommabopligi suniy intellekt usullari asosida bashorat qilinadi. Fayl o'lchamlari har xil bo'lishi mumkinligi sababli, saqlangan fayllar soni ham farq qilishi mumkin. Agar yaqin vaqt oralig'ida so'ralgan fayllar ro'yxati juda katta bo'lsa, faqat $x_{if}(t)$ ning eng katta qiymatlariga mos keladigan fayllar kesh chegaralariga joylashtiriladi.

Ishlab chiqilgan algoritmda foydalanuvchi tomonidan kelib tushgan so'rovlar asosida kesh server video kontentni taqdim etadi. Bunda agar so'ralgan kontent keshda mavjud bo'lsa (1) keshdan, aks holatda markaziy serverdan kesh orqali strimming tarzida (2) video taqdim etiladi. Har ikkala holatda ham keshda mavjud fayllarning oqimlari kutilmalari $x_{if}(t)$ qiymatlari aniqlab boriladi. Bir vaqtning

o‘zida kesh serverdagi video atributlari va ichki fayllar atributlari asosida markaziy serverlarda kontent ommabopligini bashorat qilish algoritmlari ishlaydi (3) hamda mavjud fayllarning ehtimoliy ommabop bo‘lishlik koeffitsiyentlari K_m ni hisoblaydi.

Kesh serverlar oldindan kiritilgan Tv up vaqt davomida fayllar zaxirasini yangilamagan holatda ishlaydi, belgilangan vaqtda esa ommabop bo‘lish ehtimolligi eng yuqori videolarni yuklab olish so‘rovini yuboradi (4). Bunda kesh hajmidan kelib chiqqan holda ommabopligi keshdagi fayllarning $x_{if}(t)$ qiymatidan yuqori bo‘lgan fayllar yuklab olinadi va $K_m > x_{if}(t)$ sharti asosida kesh fayllari almashtiriladi (5).

Xulosa

Ushbu ishdagi algoritm kesh serverda foydalanuvchilar tomonidan ko‘plab marotaba so‘ralishi mumkin bo‘lgan video kontentning joylashtirilishini hamda kelajakdagi video so‘rovlarga maksimal darajada chegaraviy keshlar asosida xizmat ko‘rsatishni ta‘minlaydi. Natijada foydalanuvchilar va markaziy serverlar o‘rtasida ma‘lumot almashish hajmi minimallashtiriladi. Bu esa o‘z navbatida Hit ratio ko‘rsatkichini oshirishga va o‘z navbatida mavjud kesh xotira imkoniyatlaridan maksimal darajada foydalanishga imkon beradi. Bu orqali markaziy serverga yoki tashqi serverlarga murojaatlar soni kamayadi va umumiy tarmoq energiya tejamkorlik oshiriladi.

Foydalanilgan adabiyotlar

1. D. Applegate, A. Archer, V. Gopalakrishnan, S. Lee, and K. Ramakrishnan, “Optimal content placement for a large-scale vod system,” IEEE/ACM Trans. Netw., vol. 24, no. 4, pp. 2114–2127, 2016.
2. X. Wang and G. B. Giannakis, “Resource allocation for wireless multiuser ofdm networks,” IEEE Trans. Inf. Theory, vol. 57, no. 7, pp. 4359–4372, 2011.
3. T. Chen, A. Mokhtari, X. Wang, A. Ribeiro, and G. B. Giannakis, “Stochastic averaging for constrained optimization with application to online resource allocation,” IEEE Trans. Signal Processing, 2017.
4. Y. Elkhatib, M. Mu, and N. Race, “Dataset on usage of a live & vod p2p iptv service,” in Proc IEEE P2P, 2014, pp. 1–5.
5. M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, “Analyzing the video popularity characteristics of large-scale user generated content systems,” IEEE/ACM Trans. Netw., vol. 17, no. 5, pp. 1357–1370, 2009.