Recomendação de vídeo e política de cache cientes de recursos dos dispositivos dos usuários

Ana Claudia B. L. Monção[1], **Sand L. Correa**[1], **Kleber V. Cardoso**[1]

[1]Instituto de Informática – Universidade Federal de Goiás (UFG)
Goiânia – GO – Brazil

{anaclaudia,sand,kleber}@inf.ufg.br

Abstract. *Recently, the coupling between video recommendation algorithms and cache maintenance policies has shown promising results, both to improve the user experience and to optimize the use of infrastructure resources. However, the features available on users' devices have not been taken into account in this approach. In this work, we introduce this concept by modifying a model and a metric from the literature in order to capture the resolution limitation of each user device, as well as the introduction of a new metric to represent user satisfaction with the recommendation. We evaluate our proposal and compare it with another state-of-the-art solution, using the database of the MovieLens project. With the exception of the scenario where all videos are of low quality, our proposal has a higher cache hit rate in most storage size configurations. Additionally, our proposal significantly improves the user experience by allowing them to enjoy the highest quality available on their devices.*

Resumo. *Recentemente, o acoplamento entre algoritmos de recomendação de vídeo e políticas de manutenção de cache tem mostrado resultados promissores, tanto para melhorar a experiência dos usuários quanto para otimizar o uso de recursos da infraestrutura. No entanto, os recursos disponíveis nos dispositivos dos usuários não têm sido levado em conta nessa abordagem. Neste trabalho, introduzimos esse conceito através da modificação de um modelo e de uma métrica da literatura de forma a capturar a limitação de resolução de cada dispositivo de usuário, assim como a introdução de uma nova métrica para representar a satisfação com a recomendação. Avaliamos a nossa proposta e comparamos com outra solução de estado-da-arte, usando a base de dados do projeto MovieLens. Com exceção do cenário onde todos os vídeos possuem baixa qualidade, a nossa proposta apresenta maior taxa de acerto da cache na maioria das configurações de tamanho de armazenamento. Além disso, nossa proposta melhora de maneira significativa a experiência dos usuários ao permitir que eles usufruam da maior qualidade disponível em seus dispositivos.*

1. Introdução

De acordo com previsões apresentadas pela Cisco [Cisco VNI 2020], o tráfego IP aumentará em mais de três vezes até 2023, enquanto o número de dispositivos conectados à Internet atingirá o número de 29,3 bilhões. Ainda de acordo com essas estimativas, 82% de todo o tráfego IP será de vídeo, sendo grande parte desse tráfego gerado por dispositivos móveis. Nesse contexto, é fundamental a adoção de soluções que melhorem (ou pelo

menos mantenham) a satisfação do usuário e também façam uso eficiente dos recursos de comunicação de dados. Conforme descreveremos a seguir, cache na borda da rede e sistemas de recomendação são ferramentas importantes para o desenvolvimento desse tipo de solução.

O uso de cache na borda da Internet é uma abordagem antiga, introduzida pelas redes de distribuição de conteúdo (*content delivery networks* – CDNs) no fim do século passado, i.e., há mais de 20 anos. Até hoje, empresas como Akamai, Cloudflare, Limelight e várias outras oferecem serviços de distribuição de conteúdo em escala global. Dada a relevância desse mercado e o impacto nas redes de comunicação, várias empresas de telecomunicações também começaram a oferecer serviço de distribuição de conteúdo, tais como AT&T, China Telecom e Telefonica. Além disso, empresas como Netflix e Google, criaram suas próprias CDNs exclusivas, conhecidas como Netflix Open Connect e Google Global Cache, respectivamente. A tecnologia *Multi-access Edge Computing* (MEC) [ETSI MEC] traz algumas novidades para esse contexto.

Inicialmente, o padrão aberto ETSI MEC [ETSI MEC] traz a oportunidade para que um maior número de empresas de infraestrutura possam oferecer serviço de computação e comunicação na borda da rede, o qual também passa a estar acessível para um maior número de provedores de serviço. Além disso, MEC não se restringe a oferecer a capacidade de armazenamento e processamento básico de CDNs clássicas, é possível utilizar funcionalidades avançadas de processamento (similares às encontradas em sistemas de Nuvem) e também obter informações e controlar recursos de rede (como largura de banda), próximos aos usuários. Ou seja, MEC foi planejada não apenas para permitir a execução de serviços tradicionais como cache de vídeo, CDNs em geral e jogos online, mas também novos serviços de realidade virtual, aumentada e mista, carros autônomos e automação industrial.

Para atender diferentes serviços e aplicações, os provedores de infraestrutura precisam implantar os servidores MEC em diferentes pontos da sua rede [Contreras et al. 2020], conforme ilustrado na Figura 1. A princípio, os serviços podem usufruir dos servidores MEC que estão em qualquer parte da infraestrutura. No entanto, à medida que se aproximam dos usuários móveis, os recursos MEC tendem a se tornar cada vez mais escassos e caros. Portanto, os servidores MEC disponíveis nas estações-base sem fio tendem a ser alocados para aplicações com restrições mais severas de latência, como jogos online e realidade aumentada [Malandrino et al. 2020]. Enquanto aplicações menos exigentes, como vídeo armazenado em fluxo contínuo (e.g., Netflix, Hulu, YouTube), podem ser atendidas adequadamente por servidores MEC disponíveis no núcleo que serve a rede de acesso. No núcleo, esses servidores ainda oferecem uma latência consideravelmente inferior à de centros de dados que precisam ser acessados através da Internet. Ainda que, para os clientes de vídeo acessando a cache, o maior benefício seja evitar os vários enlaces envolvidos em uma comunicação que depende da Internet, assim como seus eventuais congestionamentos e falhas que afetam a qualidade de experiência (QoE).

Por outro lado, sistemas de recomendação, cujo objetivo principal é indicar conteúdos que combinem com as preferências dos usuários, se tornaram componentes fundamentais em serviços de provisão de conteúdo. Estima-se que 80% do tempo transmitido pela Netflix é resultado de recomendações feitas pelo seu sistema

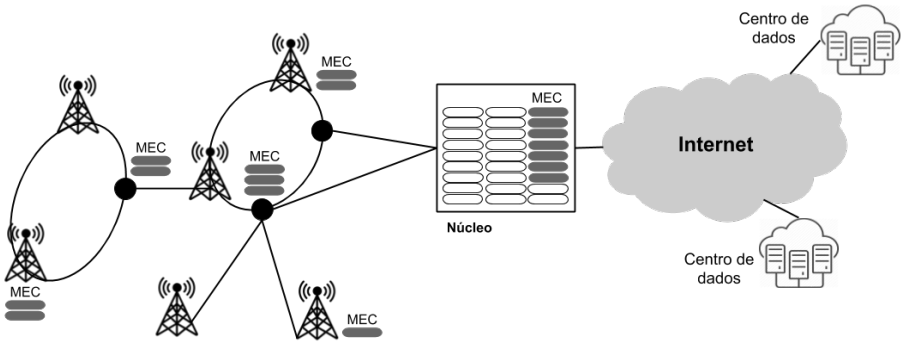


Figura 1. Servidores MEC implantados em diferentes partes da rede.

[Gomez-Uribe and Hunt 2015]. Esse fato ilustra como os sistemas de recomendação influenciam os padrões de acesso a conteúdos dos usuários. Essa influência, por outro lado, pode ser explorada para induzir padrões de acesso que beneficiam os mecanismos de cache. Recentemente, essa ideia começou a ser investigada [Chatzieleftheriou et al. 2017, Chatzieleftheriou et al. 2019] e apresenta resultados promissores, tanto para os provedores de serviço que mantêm as caches quanto para os usuários que consomem os vídeos.

Em [Chatzieleftheriou et al. 2019], os autores apresentam um modelo de otimização que combina armazenamento em cache e recomendação de vídeo visando maximizar a taxa de acertos da cache afetando minimamente as preferências dos usuários. Os autores assumem que todos os vídeos em cache são independentes e os tratam separadamente, desconsiderando a existência de vídeos idênticos codificados com resoluções diferentes. No entanto, para que um dispositivo seja capaz de reproduzir uma dessas versões de um vídeo, recursos como o tamanho da tela, a capacidade do processador, o tipo da placa de vídeo instalada, entre outros precisam ser considerados. Ao realizar o acoplamento da política de cache com um algoritmo de recomendação sem levar em conta as diferentes resoluções de um vídeo recomendado para um dispositivo móvel, a solução de [Chatzieleftheriou et al. 2019] perde oportunidade de elevar a taxa de acerto da cache e de satisfazer aos usuários aproveitando os recursos disponíveis em seus dispositivos, conforme ilustraremos na Seção 6.

Neste trabalho, adaptamos o modelo proposto em [Chatzieleftheriou et al. 2019] para resolver esse problema; modificamos o cálculo da taxa de acerto da cache para levar em conta também a limitação de resolução de cada dispositivo móvel; e introduzimos uma métrica que avalia a qualidade da recomendação de acordo com o desejo do usuário de usufruir ao máximo de seu aparelho. Chamamos nossa solução de *Resource-Aware Video Recommendation* (RAViR). Avaliamos o RAViR e comparamos com a solução proposta em [Chatzieleftheriou et al. 2019] utilizando a base de dados do projeto MovieLens [Harper and Konstan 2015]. Nossa proposta supera a rival na taxa de acerto da cache sempre que vídeos de melhores resoluções são recomendados e consegue atingir o máximo da satisfação dos usuários com relação à qualidade dessas recomendações.

O artigo está organizado da seguinte forma: Seção 2, uma breve revisão de trabalhos relacionados; Seção 3 e Seção 4, o modelo do sistema e o modelo de recomendação; Seção 5, a formulação do problema a ser resolvido; Seção 6, a exposição e avaliação dos resultados; e finalmente na Seção 7, as considerações finais.

2. Trabalhos Relacionados

Nesta seção, apresentamos alguns trabalhos relacionados com o tema principal deste artigo. Inicialmente, discutimos alguns artigos que mostram o uso de tecnologias na borda da rede, principalmente quando se refere a cache. Em seguida, apresentamos trabalhos que mostram a relação entre a qualidade do serviço e a experiência do usuário, e o uso da cache na borda quando se trata de transmissão de vídeos (streaming). Por fim, discutimos trabalhos relacionados a integração entre sistemas de recomendação e políticas de cache com a finalidade de melhorar a QoE.

Com relação ao uso de tecnologias de rede que permitem serviços de TI na borda, [Tran et al. 2016] e [Ndikumana et al. 2019] apresentam um modelo de redes de servidores MEC em um espaço colaborativo compartilhando cache, processamento, comunicação e controle para atender aos usuários evitando requisições para o servidor central, e melhorando a experiência do usuário. Os vídeos e suas versões ficam distribuídas entre os servidores MEC e, baseado em uma requisição do usuário, o modelo identifica a melhor estratégia para entregar o conteúdo e manter as caches na borda da rede. [Sasikumar et al. 2019] propõe uma estratégia de distribuição das versões dos vídeos em servidores de cache nas bordas da rede, de acordo com os interesses dos dispositivos dos usuários buscando maximizar a utilidade da rede e garantir que os usuários sempre consigam obter os vídeos selecionados nos servidores de cache.

Quanto à preocupação com a qualidade na transmissão de conteúdos, [Sermpezis et al. 2019] mostra que a satisfação do usuário não é só pelo conteúdo interessante mas também pela qualidade com que ele é transmitido. Em [Nam et al. 2016], os autores desenvolveram uma ferramenta para análise de QoE que identifica eventos, durante a reprodução de um vídeo do YouTube, que levam um usuário a abandonar o vídeo. As recargas durante a reprodução e as constantes alterações de taxas de bits são as principais causas. [Carvalho et al. 2019] apresenta uma solução usando aprendizado de máquina para predizer a resolução nos dispositivos dos usuários e criar grupos de dispositivos (D2D) de mesmas resoluções. Os autores propõem um orquestrador de rede que gerencia esses grupos e tenta manter uma qualidade na transmissão dos vídeos, reduzindo o acesso à rede 4G.

Considerando o uso de políticas de cache baseadas em sistemas de recomendação, [Sermpezis et al. 2019] mostra evidências oriundas de experimentos onde pequenas alterações em recomendações para otimizar a cache não são percebidas como intrusivas. Quando, por exemplo, um vídeo de melhor qualidade é recomendado em detrimento a outro de maior interesse e pior qualidade. Em [Qi et al. 2018], os autores combinam otimização da cache e políticas de recomendação para melhorar a qualidade da transmissão e a satisfação do usuário avaliando diferentes tipos de transmissão.

Entre os trabalhos que combinam políticas de cache e sistemas de recomendação, o trabalho proposto em [Chatzieleftheriou et al. 2019] é o mais próximo à proposta deste artigo. Explora a influência de sistema de recomendação na escolha dos usuários para induzir um perfil de tráfego mais amigável para os mecanismos de cache. A partir de uma lista ordenada das preferências dos usuários pelos vídeos do catálogo, são calculadas as probabilidades dos usuários solicitarem cada vídeo. Usando essas probabilidades, o problema *JCRP (Joint Caching and Recommendation Problem)* otimiza a cache para atender uma grande porção da demanda de usuários servidos por ela, aumentando a QoE,

diminuindo o uso dos links de comunicação e, consequentemente, a latência na entrega dos vídeos. A taxa de acerto da cache foi usada para avaliação dos resultados, os quais demonstraram que o método proposto trouxe ganhos relevantes no desempenho da cache sem degradar significativamente a preferência dos usuários.

A principal diferença entre a nossa proposta e a apresentada em [Chatzieleftheriou et al. 2019] está na preocupação com a qualidade do vídeo a ser entregue para o usuário. As recomendações passam a se referir não apenas a um vídeo cujo conteúdo é desejado pelo usuário mas que tenha a melhor resolução possível buscando usufruir o máximo dos recursos de cada dispositivo. Até o momento, não encontramos trabalhos similares com sistemas de recomendação e cache que levem em conta características do dispositivo móvel.

3. Modelo do Sistema

Neste trabalho, consideramos o ambiente MEC ilustrado na Figura 2. Múltiplas estações-base, em diferentes localidades, oferecem serviço de conectividade a um conjunto $\mathcal{U}$ de usuários de dispositivos móveis. Esses usuários utilizam diferentes tipos de dispositivos, com recursos (e.g. processadores, placa gráfica, memória e tamanho de tela) diferentes, para consumir serviços de vídeos armazenado em fluxo contínuo. Servidores MEC implantados no núcleo que serve a rede de acesso hospedam um conjunto $\mathcal{C}$ de caches. Cada cache $n \in \mathcal{C}$ tem uma capacidade de armazenamento limitada, denotada por C_n e, num dado instante no tempo, armazena um subconjunto P_n de todo conteúdo disponível. Adicionalmente, servidores em centros de dados remotos hospedam cópias de todo o catálogo de vídeos, inclusive vídeos com mesmo conteúdo mas codificados com resoluções diferentes.

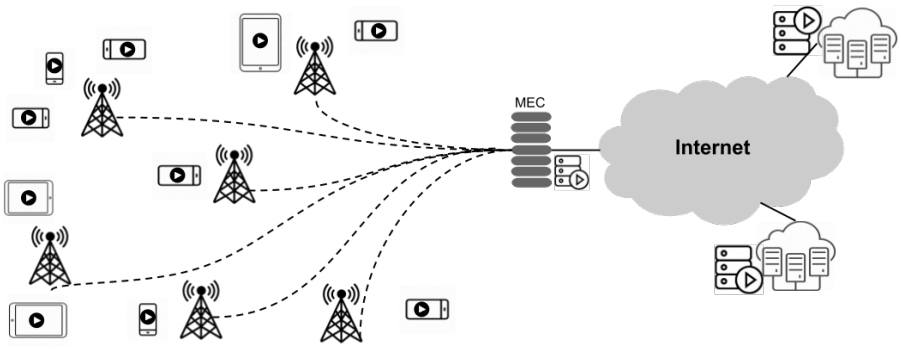


Figura 2. Cenário de referência para o modelo do sistema.

O catálogo consiste de um conjunto de vídeos $\mathcal{V}$ onde cada vídeo $v \in \mathcal{V}$ é identificado pelo seu conteúdo e sua resolução. Seja $\mathcal{Q} = \{1, 2, ..., |\mathcal{Q}|\}$ o conjunto de todas as resoluções possíveis, sendo 1 a resolução mais baixa e $|\mathcal{Q}|$ a mais alta. Vídeos codificados com resoluções diferentes são considerados itens diferentes no catálogo e têm tamanhos diferentes. Denotamos por q_v e tam_v a resolução e o tamanho do vídeo $v \in \mathcal{V}$.

Seja $\mathcal{M} = \{cat_1, cat_2, ..., cat_{|\mathcal{M}|}\}$ o conjunto de categorias temáticas disponíveis no catálogo. Representamos por f^v o vetor de característica temáticas associado ao vídeo v, onde $f^v(cat_j)$, $j \in \{1, 2, ..., |\mathcal{M}|\}$, representa a relevância da categoria cat_j para o vídeo v. Essa relevância assume valores no intervalo [0,1] sendo normalizada, de forma que $\sum_{j=1}^{|\mathcal{M}|} f^v(cat_j) = 1, \forall v \in \mathcal{V}$.

Cada usuário $u \in \mathcal{U}$, em determinado período de tempo, é atendido por uma cache usando um dispositivo móvel que aceita um limite máximo de resolução q_u. De forma similar aos vídeos, um usuário u é representado por um vetor de características temáticas f^u, calculado com base nos vídeos assistidos e avaliados por ele. Cada elemento $f^u(cat_j)$, $j \in \{1, 2, ..., |\mathcal{M}|\}$ representa o grau de interesse do usuário u pela categoria temática cat_j. Esse interesse assume valores no intervalo [0,1], sendo normalizado de forma que $\sum_{j=1}^{|\mathcal{M}|} f^u(cat_j) = 1, \forall u \in \mathcal{U}$,

Sendo a representação dos vídeos (f^v) e a representação dos usuários (f^u) vetores que indicam o grau de relevância em cada categoria temática, utilizamos a similaridade de cossenos para identificar o interesse de um usuário pelo conteúdo de um determinado vídeo, conforme sugerido em [Chatzieleftheriou et al. 2019]. Essa similaridade é definida por:

$$sim_{u,v} = \frac{\sum_{j=1}^{|\mathcal{M}|} f^u(cat_j) \cdot f^v(cat_j)}{\sqrt{\sum_{j=1}^{|\mathcal{M}|} f^u(cat_j)^2} \cdot \sqrt{\sum_{j=1}^{|\mathcal{M}|} f^v(cat_j)^2}}. \quad (1)$$

Baseado na similaridade, para cada usuário $u \in \mathcal{U}$ e cada vídeo do catálogo $v \in \mathcal{V}$, podemos calcular uma distribuição de preferências de conteúdo dada por:

$$p_u^{pCont}(v) = \frac{sim_{u,v}}{\sum_{v \in \mathcal{V}} sim_{u,v}}, \quad (2)$$

onde $\sum_{v \in \mathcal{V}} p_u^{pCont}(v) = 1$.

4. Modelo de Recomendação

A literatura demonstra que sistemas de recomendação têm um grande impacto nas escolhas de um usuário. Em geral, eles aumentam a demanda para os itens recomendados enquanto diminuem proporcionalmente a demanda pelos demais itens de um catálogo [Zhou et al. 2010, Krishnappa et al. 2015]. Além disso, há fortes evidências que a quantidade de itens recomendados e suas respectivas posições na lista de recomendação também influenciam a escolha do usuário.

Para capturar a forma como as recomendações direcionam a escolha do usuário, em [Chatzieleftheriou et al. 2019], os autores assumem que uma distribuição $p_u^{rec}(v)$ impulsiona igualmente todos os vídeos v recomendados para um usuário u, ou seja:

$$p_u^{rec}(v) = 1/R, \quad (3)$$

onde R representa a quantidade de vídeos recomendados para o usuário u. A intuição por trás da Equação 3 é que quanto menor for a lista de vídeos recomendados, maior será a influência que a recomendação terá na escolha do usuário. Isso é especialmente verdade para usuários móveis, onde os dispositivos com telas pequenas dificultam a rolagem para a escolha.

Os autores então definem um distribuição de solicitação de vídeos como sendo uma combinação das distribuições $p_u^{pCont}(v)$ e $p_u^{rec}(v)$, da seguinte forma. Para os R vídeos recomendados, essa distribuição é calculada como:

$$p_u^{solR}(v) = p_u^r \cdot p_u^{rec}(v) + (1 - p_u^r) \cdot p_u^{pCont}(v), \quad (4)$$

onde p_u^r é o peso que expressa a importância que o usuário u dá para uma recomendação.

Para os demais $(|\mathcal{V}| - R)$ vídeos não recomendados do catálogo, a distribuição de solicitação de vídeos é definida por:

$$p_u^{solR\sim}(v) = (1 - p_u^r) \cdot p_u^{pCont}(v). \tag{5}$$

Explorando o potencial de um sistema de recomendação como modelador de solicitações de vídeo, os autores em [Chatzieleftheriou et al. 2019] propõem o uso desse potencial para otimizar políticas de cache de conteúdo. Para isso, ao invés de emitirem recomendações para os R primeiros vídeos da lista de recomendação de um usuário, como normalmente esperado para um sistema de recomendação, os autores propõem selecionar R vídeos em uma janela de recomendação ampliada W_u. Essa janela é definida sobre os K_u primeiros itens da lista de recomendação, onde $K_u > R$. K_u é definido de acordo com uma tolerância de distorção $tol_u \in [0, 1)$ que indica um limite aceito pelo usuário sobre suas preferências. Considerando uma lista W_u de K_u vídeos, no pior caso os últimos R vídeos serão recomendados, ou seja, os vídeos das posições $K_u - R + 1, K_u - R + 2, .., K_u$. Seja $pos_u(v)$ a posição de um vídeo v na lista de recomendação do usuário u, a distorção no pior caso é definida como:

$$\Delta_u(K_u, R) = 1 - \frac{\sum_{j:pos_u(v)\in[K_u-R+1,K_u]} p_u^{pCont}(v)}{\sum_{j:pos_u(v)\in[1,R]} p_u^{pCont}(v)}. \tag{6}$$

Baseado na distorção do pior caso, $\Delta_u(K_u, R)$, e na tolerância $tol_u \in [0, 1)$, a cardinalidade K_u da janela de recomendação ampliada W_u é dada por:

$$K_u = max\{k|\Delta_u(K_u, R) \leq tol_u)\}. \tag{7}$$

5. Combinando Política de Cache com Sistemas de Recomendação cientes de Recurso

No problema formulado em [Chatzieleftheriou et al. 2019], a distribuição $p_u^{pCont}(v)$ (Equação 2) descreve as preferências de conteúdo de um usuário. Essa distribuição, por sua vez, considera apenas o interesse do usuário pelas características temáticas do vídeo. Por outro lado, como a qualidade do serviço é predominante para a satisfação e engajamento do usuário com o conteúdo [Sermpezis et al. 2019], a resolução em que um vídeo é reproduzido pode afetar positivamente ou negativamente a preferência do usuário. Como consequência, vídeos com uma resolução que o dispositivo do usuário não consiga reproduzir não devem constar entre as suas preferências, mesmo que o conteúdo seja interessante.

Para refletir essa situação, neste trabalho, propomos o fator $int_{u,q}$, cujo valor varia no intervalo [0,1] e que representa a relevância de uma resolução q para o dispositivo de um usuário u, sendo seu valor dado por:

$$int_{u,q} = \begin{cases} 0, & q_u < q \\ 1, & q_u = q \\ 1 - (q_u - q)/q_u, & q_u > q. \end{cases} \tag{8}$$

De acordo com a Equação 8, a resolução mais apropriada para o usuário terá o fator $int_{u,q}$ igual a 1 e as demais resoluções terão uma redução progressiva relacionada com a expectativa do usuário. Resoluções que não podem ser reproduzidas pelo dispositivo do usuário terão o fator $int_{u,q}$ igual a 0.

Para aplicar o fator $int_{u,q}$ a cada vídeo v, definimos uma nova distribuição que descreve as preferências dos usuários pelas resoluções dos vídeos, representada por:

$$p_u^{pRes}(v) = \frac{int_{u,q_v}}{\sum_{v \in \mathcal{V}} int_{u,q_v}}, \tag{9}$$

onde q_v é a resolução do vídeo v e $\sum_{v \in \mathcal{V}} p_u^{pRes}(v) = 1$.

Combinando as preferências de conteúdo (p_u^{pCont}) com as preferências de resolução (p_u^{pRes}), definimos uma nova distribuição, chamada de preferências qualitativas, relacionada com a experiência do usuário em receber um vídeo com um conteúdo interessante e com a melhor resolução possível diante das limitações do seu dispositivo. Essa distribuição é dada por:

$$p_u^{pQual}(v) = \frac{p_u^{pCont}(v) \cdot p_u^{pRes}(v)}{\sum_{v \in \mathcal{V}} p_u^{pCont}(v) \cdot p_u^{pRes}(v)}, \tag{10}$$

onde $\sum_{v \in \mathcal{V}} p_u^{pQual}(v) = 1$.

Redefinimos então a distribuição de solicitação de vídeos, representada pelas Equações 4 e 5 na proposta de [Chatzieleftheriou et al. 2019], substituindo as preferências de conteúdo (p_u^{pCont}) do usuário pelas suas preferências qualitativas ($p_u^{pQual}(v)$), ou seja:

$$p_u^{solR}(v) = p_u^r \cdot p_u^{rec}(v) + (1 - p_u^r) \cdot p_u^{pQual}(v), \tag{11}$$

para os R vídeos recomendados e

$$p_u^{solR\sim}(v) = (1 - p_u^r) \cdot p_u^{pQual}(v), \tag{12}$$

para os demais $(|\mathcal{V}| - R)$ vídeos não recomendados do catálogo.

Redefinimos também a Equação 6 de forma que o tamanho das listas de recomendações ampliadas dos usuários W_u possa refletir a nova distribuição de preferências qualitativas, ou seja:

$$\Delta_u(K_u, R) = 1 - \frac{\sum_{j:pos_u(v) \in [K_u - R + 1, K_u]} p_u^{pQual}(v)}{\sum_{j:pos_u(v) \in [1,R]} p_u^{pQual}(v)}. \tag{13}$$

Definimos então uma nova política de cache e recomendação, denominada *Resource-Aware Video Recommendation* (RAViR), cujo objetivo é maximizar a taxa de acerto da cache para um determinado grupo de usuários, usando os sistemas de recomendação como meio de identificar e impulsionar as preferências dos usuários com a consciência dos recursos disponíveis em cada dispositivo.

Formalmente, sejam y_v e $x_{u,v}$, $v \in \mathcal{V}, u \in \mathcal{U}$, duas variáveis de decisão. A primeira (y_v) indica se um vídeo v está em cache ($y_v = 1$) ou não ($y_v = 0$), enquanto

Tabela 1. Tabela de Notações

Notação	Significado
$\mathcal{V}$	Conjunto de vídeos do catálogo
$\mathcal{U}$	Conjunto de usuários
$\mathcal{M}$	Conjunto de categorias temáticas
$\mathcal{Q}$	Conjunto de Resoluções
tam_v	Tamanho do vídeo
q_v	Qualidade/Resolução do vídeo
q_u	Máxima resolução aceita pelo equipamento do vídeo
f^v	Vetor de categorias temáticas que representa um vídeo
f^u	Vetor de categorias temáticas que representa um usuário
$sim_{u,v}$	Similaridade entre um usuário e um vídeo
$int_{u,q}$	Interesse de um usuário por uma resolução
p_u^{pCont}	Distribuição de preferência por conteúdo
p_u^{pRes}	Distribuição de preferência por resolução
p_u^{pQual}	Distribuição de preferência qualitativa (Conteúdo e Resolução)
p_u^{rec}	Distribuição de probabilidade devido à recomendação
p_u^{solR}	Distribuição de probabilidade de solicitação de vídeos recomendados
$p_u^{solR\sim}$	Distribuição de probabilidade de solicitação de vídeos não recomendados
p_u^r	Peso que o usuário dá para uma recomendação
C_n	Capacidade da cache n
P_n	Conteúdo da cache n
R	Quantidade de recomendações por usuário
W_u	Janela de recomendação do usuário
K_u	Tamanho da janela W_u
tol_u	Tolerância de distorção sobre a recomendação aceita pelo usuário

a segunda ($x_{u,v}$) indica se um vídeo v está entre os recomendados para um usuário u ($x_{u,v} = 1$) ou não ($x_{u,v} = 0$). O objetivo da nossa solução, (RAViR), é formulado como:

$$\max_{y,x} \sum_{u \in \mathcal{U}} \sum_{v \in W_u} y_v (x_{u,v} \cdot p_u^{solR}(v) + (1 - x_{u,v}) \cdot p_u^{solR\sim}(v)) \tag{14}$$

$$s.t. \sum_{v \in \mathcal{V}} y_v \cdot tam_v \leq C_n \tag{15}$$

$$\sum_{v \in W_u} x_{u,v} = R, \forall u \in \mathcal{U} \tag{16}$$

$$p_u^{pRes}(v) > 0, \forall u \in \mathcal{U}, \forall v \in W_u \tag{17}$$

$$y_v, x_{u,v} \in \{0, 1\}, \forall u \in \mathcal{U}, \forall v \in W_u. \tag{18}$$

A Equação 14 representa a maximização da taxa de acerto da cache usando a lista de preferências de cada usuário W_u. As restrições apresentadas são referentes à capacidade da cache (Equação 15), à quantidade de vídeos a serem recomendados (Equação 16)

e à recomendação de um vídeo ser compatível com o dispositivo do usuário (Equação 17). A Tabela 1 apresenta um resumo das notações usadas nas Seções 3, 4 e 5.

6. Experimentos e Avaliação dos Resultados

Para avaliarmos nossa política de cache e recomendação RAViR, utilizamos a mesma base de dados utilizada em [Chatzieleftheriou et al. 2019]. Essa base consiste em um subconjunto do projeto MovieLens [Harper and Konstan 2015], com 610 usuários ($|\mathcal{U}| = 610$) e 9.742 vídeos ($|\mathcal{V}| = 9.742$) descritos por 19 categorias temáticas ($|\mathcal{M}| = 19$). Como consideramos versões com resoluções diferentes para cada vídeo, foi necessário alterarmos o catálogo para acrescentar as versões, gerando um novo catálogo com 24.347 vídeos ($|\mathcal{V}| = 24.347$).

Consideramos inicialmente um conjunto de 4 tipos de resoluções ($|\mathcal{Q}| = 4$) conforme definido pela Netflix e por [Nandakumar et al. 2019], sendo 1-SD (480p), 2-HD (720p), 3-Full HD (1080p) e 4-Ultra HD (2160p). Para cada vídeo do catálogo, geramos um valor aleatório entre [1,4] representando a maior resolução disponível para o vídeo. A partir desse valor, acrescentamos ao catálogo as versões do vídeo com todas as resoluções até esse limite. Ou seja, se o valor gerado aleatoriamente foi 4, acrescentamos ao catálogo as versões do mesmo vídeo com resoluções 1, 2, 3 e 4. O tamanho de cada vídeo na resolução 1 (480p) foi gerado por uma distribuição uniforme $U(1; 4)$. Para as demais resoluções, aplicamos fatores que equivalem à relação de *pixels* entre cada resolução e a menor resolução, ou seja: 1,5 para a versão com resolução 2 (720p); 2,5 para a versão com resolução 3 (1080p); e 3,5 para a versão com resolução 4 (2160p).

Em seguida, calculamos os vetores de características que representam os usuários (f^u) e os vídeos (f^v), suas similaridades ($sim_{u,v}$) e os interesses pelas resoluções ($int_{u,q}$). Para isto, usamos informações que os provedores têm sobre os dispositivos dos usuários. Assim como em [Chatzieleftheriou et al. 2019], geramos o peso p_u^r, relacionado à importância que cada usuário dá para as recomendações, usando uma distribuição uniforme $U(0,5; 0,7)$.

Avaliamos os resultados obtidos pelo RAViR comparando-o com o modelo JCRP proposto em [Chatzieleftheriou et al. 2019]. Como o catálogo utilizado consiste de várias versões do mesmo vídeo, não previsto no modelo JCRP, foi preciso fazer alguns ajustes para os experimentos. No modelo JCRP as preferências pelas versões do mesmo vídeo são iguais pois consideram apenas o interesse pelo conteúdo (p^{pCont}). Assim, para termos diferentes avaliações e não prejudicarmos o JCRP, fizemos experimentos com derivações do modelo JCRP, restringindo as versões no momento de gerar a lista W_u, da seguinte forma:

- JCRP-L: considera as versões de menor resolução para serem armazenadas na cache e recomendadas;
- JCRP-M: considera as versões de média resolução para serem armazenadas na cache e recomendadas;
- JCRP-H: considera as versões de alta resolução para serem armazenadas na cache e recomendadas.

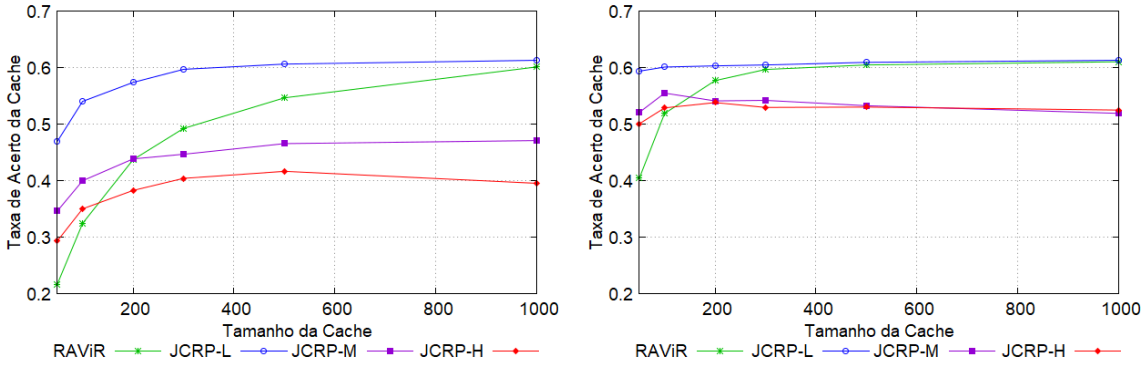
Para a realização dos experimentos, consideramos a quantidade de 3 recomendações ($R = 3$) e variamos a tolerância (tol_u) e o tamanho da cache (C_n). Para o tamanho

da cache utilizamos: 50, 100, 200, 300, 400, 500 e 1000. Para a tolerância, consideramos: 0,01 e 0,1. O primeiro valor de tolerância resulta em uma lista contendo vídeos com preferências muito próximas e, consequentemente, limita bastante a quantidade de vídeos na lista W_u dos usuários. O segundo valor de tolerância gera uma lista de preferências maior, com valores mais esparsos e as recomendações tornam-se mais flexíveis.

Como em [Chatzieleftheriou et al. 2019], usamos a taxa de acerto da cache para a comparação do RAViR com as diferentes variantes do JCRP. No entanto, no contexto deste trabalho, que trata a resolução como uma característica do vídeo, reformulamos a métrica de acerto da cache para desprezar a probabilidade de um vídeo na cache ser requisitado por um usuário quando o seu dispositivo não tem recursos suficientes para a reprodução do vídeo. Para isso, acrescentamos uma variável binária $z_{u,v}$ com valor 1 caso o usuário u esteja apto a assistir o vídeo v, ou valor 0 caso contrário, ou seja:

$$H = \frac{\sum_{u \in \mathcal{U}} \sum_{v \in P} z_{u,v} \cdot p_u^{solR}(v)}{\sum_{u \in U} \sum_{v \in V} p_u^{solR}(v)}. \tag{19}$$

A Figura 3 apresenta como a taxa de acerto da cache se comporta à medida que a capacidade de armazenamento da cache aumenta para as diferentes soluções. A Figura 3(a) apresenta os resultados para a tolerâncias igual a 0,01, enquanto a figura 3(b) mostra os resultados para a tolerância 0,1.



(a) $|\mathcal{U}| = 610$, $|\mathcal{V}| = 24.347$, $R = 3$ e $tol = 0,01$ (b) $|\mathcal{U}| = 610$, $|\mathcal{V}| = 24.347$, $R = 3$ e $tol = 0,1$

Figura 3. Comparação entre RAViR e JCRP

Para a tolerância 0,01, observamos que a solução JCRP-L tem desempenho igual ou superior ao RAViR para todos os tamanhos da cache. Isso ocorre porque no JCRP-L, todos os vídeos são menores e de baixa resolução sendo possível armazenar mais vídeos na cache. Além disso, no JCRP-L as versões dos vídeos têm preferências iguais pois não considera a resolução. Logo, qualquer versão recomendada tem o mesmo impacto na taxa de acerto da cache pois mais usuários compartilham as mesmas preferências, mesmo com dispositivos diferentes, possibilitando a recomendação dos mesmos vídeos para vários usuários. Como desvantagem, o JCRP-L só consegue recomendar versões com baixa resolução diminuindo a qualidade dos vídeos recomendados e a satisfação do usuário. As soluções JCRP-M e JCRP-H inicialmente apresentam um desempenho melhor pois também têm os valores de preferências iguais para todas as versões de um mesmo vídeo e, portanto, conseguem recomendar as mesmas versões para mais usuários, mesmo com

dispositivos diferentes. Entretanto, como não consideram a compatibilidade com a resolução, podem recomendar versões que o usuário não consegue reproduzir, caindo muito a taxa de acerto da cache, o que não acontece com o RAViR, o qual sempre recomenda vídeos com resoluções compatíveis. Com o tamanho da cache muito pequeno, o RAViR tem um desempenho inferior pois as versões do mesmo vídeo têm valores diferentes de preferências e nem sempre são compatíveis com muitos usuários. Isso dificulta a recomendação de versões iguais para vários usuários, diminuindo a taxa de acerto da cache. À medida que o tamanho da cache aumenta, o RAViR consegue um desempenho superior e crescente, garantindo que o vídeo recomendado para o usuário tenha a melhor resolução possível e seja compatível com o seu dispositivo, evitando procedimentos de transcodificação que exigem mais recursos dos provedores.

Para a tolerância 0,1, os resultados apresentam algumas diferenças. A taxa de acerto da cache começa com valores superiores para todas as soluções mas, à medida que o tamanho da cache aumenta, torna-se possível armazenar vídeos com tamanhos maiores. Nessa situação, as soluções derivadas do JCRP fazem escolhas por vídeos com tamanhos maiores sem levar em consideração as compatibilidades com os dispositivos, o que reduz a taxa de acerto da cache.

Considerando que cada usuário espera usufruir dos recursos disponíveis no seu equipamento, introduzimos uma métrica baseada no fator int_{u,q_v} (Equação 8) que representa a relevância de cada resolução para um dispositivo, tendo valor 1 quando o vídeo tem a resolução esperada pelo usuário, ou seja, a melhor qualidade possível para ser reproduzida com os recursos disponíveis no seu dispositivo. Assim, a métrica foi criada para avaliar o quanto cada método satisfaz as expectativas dos usuários com relação à qualidade das recomendações:

$$S = \frac{\sum_{u \in \mathcal{U}} \sum_{v \in \mathcal{V}} x_{u,v} \cdot int_{u,q_v}}{|U| \cdot R}. \tag{20}$$

A Figura 4 mostra as médias dos índices de qualidade das recomendações (Equação 20) obtidos nos experimentos de cada política de cache e recomendação. Podemos observar que o RAViR sempre consegue atingir 100% das expectativas dos usuários nesse quesito, ou seja, sempre recomenda vídeos com a qualidade esperada pelo usuário. Já os demais métodos apresentam resultados inferiores.

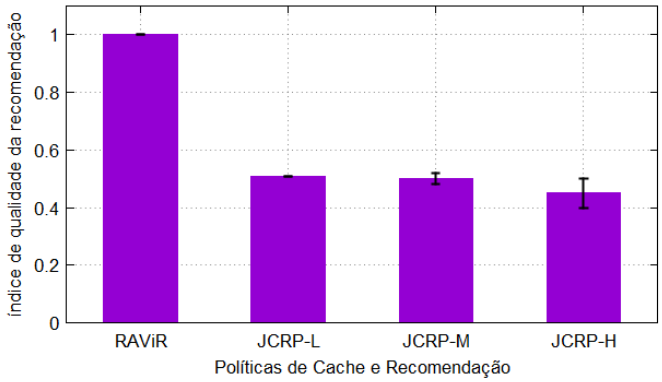


Figura 4. Satisfação com a Qualidade da Recomendação

Esses resultados mostram que a solução proposta (RAViR) consegue superar as soluções derivadas do JCRP com um crescimento progressivo na taxa de acerto da cache e com a garantia da recomendação de vídeos com a melhor resolução possível, dados os recursos disponíveis nos dispositivos dos usuários.

7. Conclusão

Neste artigo, buscamos otimizar conjuntamente cache e recomendações para melhorar o desempenho da rede e a satisfação do usuário. Apresentamos um modelo de otimização onde os sistemas de recomendação são conscientes dos recursos disponíveis e influenciam nas decisões de armazenamento da cache levando em conta esses recursos. Mostramos que recomendações baseadas nos conteúdos e nas resoluções dos vídeos aumentam a probabilidade de versões de vídeos serem solicitadas e consequentemente melhoram a taxa de acerto da cache e a satisfação do usuário. Os primeiros experimentos mostraram que o modelo proposto teve ganho significativo quando comparado ao de [Chatzieleftheriou et al. 2019] no que diz respeito a recomendar um conteúdo de interesse do usuário com a melhor resolução possível dentro dos recursos disponíveis.

Como um trabalho inicial nesta área de pesquisa, os resultados foram promissores mas ainda existem muitos pontos que podem ser explorados em trabalhos futuros, entre eles: o uso da largura de banda como restrição no problema para garantir que essa seja suficiente para que todos os usuários consigam receber os vídeos na melhor resolução possível aceita pelo seu dispositivo; a análise da complexidade do modelo e criação de métodos de aproximação para bases de dados maiores; a análise de pesos entre o interesse pelo conteúdo e o interesse pelas resoluções dos vídeos, assim como outras questões relacionadas ao problema.

Referências

[Carvalho et al. 2019] Carvalho, M., Silva, V. F., de Britto e Silva, E., Macedo, D. F., de Resende, H. C. C., Marquez-Barja, J. M., Both, C. B., Bardini, A. Z., and Wickboldt, J. (2019). Qoe-based video orchestration for 4g networks. In *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1–6.

[Chatzieleftheriou et al. 2017] Chatzieleftheriou, L. E., Karaliopoulos, M., and Koutsopoulos, I. (2017). Caching-aware recommendations: Nudging user preferences towards better caching performance. In *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, pages 1–9.

[Chatzieleftheriou et al. 2019] Chatzieleftheriou, L. E., Karaliopoulos, M., and Koutsopoulos, I. (2019). Jointly Optimizing Content Caching and Recommendations in Small Cell Networks. *IEEE Transactions on Mobile Computing*, 18(1):125–138.

[Cisco VNI 2020] Cisco VNI (2020). Cisco Visual Networking Index: Forecast and Trends, 2017–2022. `https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf`. [Última visita: 16-Março-2020].

[Contreras et al. 2020] Contreras, L. M., Baliosian, J., Martinez-Julia, P., and Serrat, J. (2020). Computing at the edge, but what edge? In *IEEE/IFIP Network Operations and Management Symposium (NOMS)*.

[ETSI MEC] ETSI MEC. Multi-access Edge Computing (MEC). https://www.etsi.org/technologies/multi-access-edge-computing. [Última visita: 25-Março-2020].

[Gomez-Uribe and Hunt 2015] Gomez-Uribe, C. A. and Hunt, N. (2015). The Netflix Recommender System: Algorithms, Business Value, and Innovation. *ACM Trans. Manage. Inf. Syst.*, 6(4):13:1–13:19.

[Harper and Konstan 2015] Harper, F. M. and Konstan, J. A. (2015). The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4).

[Krishnappa et al. 2015] Krishnappa, D. K., Zink, M., Griwodz, C., and Halvorsen, P. (2015). Cache-centric video recommendation: An approach to improve the efficiency of youtube caches. *ACM Trans. Multimedia Comput. Commun. Appl.*, 11(4).

[Malandrino et al. 2020] Malandrino, F., Chiasserini, C.-F., Avino, G., Malinverno, M., and Kirkpatrick, S. (2020). From Megabits to CPU Ticks: Enriching a Demand Trace in the Age of MEC.

[Nam et al. 2016] Nam, H., Kim, K., and Schulzrinne, H. (2016). Qoe matters more than qos: Why people stop watching cat videos. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9.

[Nandakumar et al. 2019] Nandakumar, D., Wu, Y., Wei, H., and Ten-Ami, A. (2019). On the accuracy of video quality measurement techniques. In *2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6.

[Ndikumana et al. 2019] Ndikumana, A., Tran, N. H., Ho, T. M., Han, Z., Saad, W., Niyato, D., and Hong, C. S. (2019). Joint communication, computation, caching, and control in big data multi-access edge computing. *IEEE Transactions on Mobile Computing*, pages 1–1.

[Qi et al. 2018] Qi, K., Chen, B., Yang, C., and Han, S. (2018). Optimizing caching and recommendation towards user satisfaction. In *2018 10th International Conference on Wireless Communications and Signal Processing (WCSP)*, pages 1–7.

[Sasikumar et al. 2019] Sasikumar, A., Zhao, T., Hou, I., and Shakkottai, S. (2019). Cache-version selection and content placement for adaptive video streaming in wireless edge networks. *CoRR*, abs/1903.12164.

[Sermpezis et al. 2019] Sermpezis, P., Kastanakis, S., Pinheiro, J. I., Assis, F., Menasché, D., and Spyropoulos, T. (2019). Towards qos-aware recommendations. *CoRR*, abs/1907.06392.

[Tran et al. 2016] Tran, T. X., Pandey, P., Hajisami, A., and Pompili, D. (2016). Collaborative multi-bitrate video caching and processing in mobile-edge computing networks. *CoRR*, abs/1612.01436.

[Zhou et al. 2010] Zhou, R., Khemmarat, S., and Gao, L. (2010). The Impact of YouTube Recommendation System on Video Views. In *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement*, pages 404–410.