Transfer Learning Approach for Cache-Enabled Wireless Networks

Ejder Baştuğ[◇], Mehdi Bennis[★] and Mérouane Debbah[◇,†]

[◇]Large Networks and Systems Group (LANEAS), CentraleSupélec, Gif-sur-Yvette, France
[★]Centre for Wireless Communications, University of Oulu, Finland
[†]Mathematical and Algorithmic Sciences Lab, Huawei France R&D, Paris, France
ejder.bastug@centralesupelec.fr, bennis@ee.oulu.fi, merouane.debbah@huawei.com

Abstract—Locally caching contents at the network edge constitutes one of the most disruptive approaches in 5G wireless networks. Reaping the benefits of edge caching hinges on solving a myriad of challenges such as how, what and when to strategically cache contents subject to storage constraints, traffic load, unknown spatio-temporal traffic demands and data sparsity. Motivated by this, we propose a novel *transfer learning*-based caching procedure carried out at each small cell base station. This is done by exploiting the rich contextual information (i.e., users' content viewing history, social ties, etc.) available from device-to-device (D2D) interactions, referred to as *source domain*. This prior information is incorporated in the so-called *target domain* where the goal is to optimally cache strategic contents at the small cells as a function of storage, estimated content popularity, traffic load and backhaul capacity. It is shown that the proposed approach overcomes the notorious data sparsity and cold-start problems, yielding significant gains in terms of users' quality-of-experience (QoE) and backhaul offloading, with gains reaching up to 22% in a setting consisting of four small cell base stations.

Index Terms—caching, transfer learning, collaborative filtering, data sparsity, cold-start problem, 5G

I. INTRODUCTION

Caching at the network edge is one of the five most promising innovations in 5G wireless networks [1]. Recently, it was shown that caching can significantly offload different segments of the infrastructure including radio access network (RAN) and core network (CN), by intelligently storing contents closer to the users. As opposed to pushing contents on a *best-effort* basis ignoring end-users' behavior and interactions, we are witnessing an era of truly context-aware and proactive networking [2]. Undoubtedly, edge caching has taken recent 5G research activities by storm as evidenced by the recent literature in both academia and industry [2]–[12] (to cite a few).

Although caching has been well-studied in wired networks, caching over wireless remains in its infancy. The idea of *femtocaching* is proposed in [3], in which small base stations (SBSs) called *helpers* with low-speed backhaul but high storage units carry out content delivery via short-range transmissions. Randomly distributed SBSs with storage capabilities

are studied in [4], characterizing the outage probability and average delivery rate. A stochastic-geometry based caching framework for device-to-device (D2D) communications is examined in [5] where mathematical expressions of local and global fractions of served content requests are given. From a game theoretic standpoint, various approaches have been studied such as multi-armed bandits under unknown content popularity [6], many-to-many matching [8] and joint content-aware user clustering and content caching [11]. Other works include information-theoretic studies looking at fundamentals of local and global caching gains in [9], facility location based approximation in [7], as well as multiple-input multiple-output (MIMO) caching in [12], and coded caching in [10].

In [2], by exploiting spatio-social caching coupled with D2D communication, we proposed a novel proactive networking paradigm in which SBSs and user terminals (UTs) proactively cache contents at the edge. As a result, the overall performance of the network in terms of users' satisfaction and backhaul offloading was improved. Therein, the *proactive* caching problem assumed non-perfect knowledge of the content popularity matrix, and supervised machine learning and collaborative filtering (CF) techniques were used to estimate the popularity matrix leveraging user-content correlations. Nevertheless, the content popularity matrix remains typically large and sparse with very few users ratings, rendering CF learning methods inefficient mainly due to *data sparseness* and *cold-start* problems [13].

Given the fact that data sparsity and cold-start problems degrade the performance of proactive caching, we leverage the framework of *transfer learning (TL)* and recent advances in machine learning [14]. TL is motivated by the fact that in many real-world applications, it is hard or even impossible to collect and label training data to build suitable prediction models. Exploiting available data from other rich information sources such as D2D interactions (called as *source domain*), allows TL to substantially improve the prediction task in the so-called *target domain*. TL has been applied to various data mining problems such as classification and regression [14]. TL methods can be mainly grouped into *inductive*, *transductive* and *unsupervised* TL methods depending on the availability of labels in the source and target domains. All these approaches boil down to answering the following fundamental questions: 1) *what* information to transfer? 2) *how* to transfer it? and 3)

when to transfer it? While "what to transfer" deals with which part of the knowledge should be transferred between domains and tasks, "when to transfer" focuses on the timing of the operations in order to avoid negative transfer, especially when the source and target domains are uncorrelated. On the other hand, "how to transfer" deals with what kind of information should be transferred between domains and tasks.

The main contribution of this work is to propose a TL-based content caching mechanism to maximize the backhaul offloading gains as a function of storage constraints and users' content popularity matrix. This is done by learning and transferring hidden latent features extracted from the source domain to the target domain. In the source domain, we take into account users' D2D interactions while accessing/sharing statistics of contents within their social community as prior information in the knowledge transfer. It is shown that the content popularity matrix estimation in the target domain can be significantly improved instead of *learning from scratch* with unknown users' ratings. To the best of our knowledge, this is perhaps the first contribution of unsupervised transfer learning in cache-enabled small cells.

The rest of the paper is organized as follows. The network model under consideration is provided in Section II, accompanied with the caching problem formulation in both source and target domains. Section III presents the classical CF-based caching and that of the proposed transfer learning. The numerical results capturing the impact of various parameters on the users' satisfaction and backhaul offloading gains are given in Section IV. We finally conclude and delineate future directions in Section V.

II. NETWORK MODEL

Let us assume an information system denoted by $S^{(\mathcal{S})}$ in the source domain and an information system denoted by $S^{(\mathcal{T})}$ in the target domain. A sketch of the network model is shown in Fig. 1.

A. Target Domain

Let us consider a network deployment consisting of M_{tar} SBSs from the set $\mathcal{M}_{tar} = \{1, \ldots, M_{tar}\}$ and N_{tar} UTs from the set $\mathcal{N}_{tar} = \{1, \ldots, N_{tar}\}$. Each SBS m is connected to the core network via a limited backhaul link with capacity $0 < C_m < \infty$ and each SBS has a total wireless link capacity C'_m for serving its UTs in the downlink. We further assume that $\mathbb{E}[C_m] < \mathbb{E}[C'_m]$. UTs request contents from a library $\mathcal{F}_{tar} = \{1, \ldots, F_{tar}\}$, where each content f has a size of $L(f)$ and a bitrate requirement of $B(f)$. Moreover, we suppose that users' content requests follow a Zipf-like distribution $P_{\mathcal{F}_{tar}}(f), \forall f \in \mathcal{F}_{tar}$ defined as [15]:

$$P_{\mathcal{F}_{tar}}(f) = \frac{\Omega}{f^\alpha} \tag{1}$$

where $\Omega = \left(\sum_{i=1}^{F_{tar}} \frac{1}{i^\alpha} \right)^{-1}$ and α characterizes the steepness of the distribution, reflecting different content popularities. Having such a content popularity in the ordered case, the content popularity matrix for the m-th SBS at time t is given
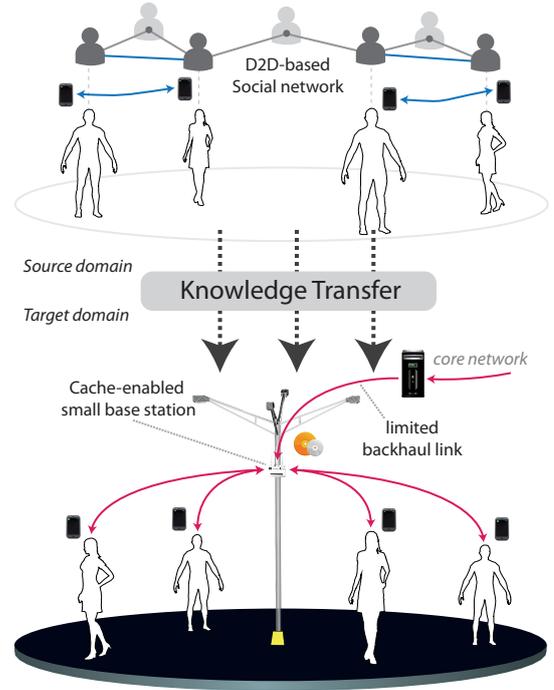

Figure 1: An illustration of the network model which consists of two information systems $S^{(\mathcal{S})}$ and $S^{(\mathcal{T})}$. Due to the lack of prior information in the target domain, the information extracted from users' social interactions and their ratings in the source domain is transferred to the target domain.

by $\mathbf{P}^m(t) \in \mathbb{R}^{N_{tar} \times F_{tar}}$ where each entry $P_{n,f}^m(t)$ represent the probability that the n-th user requests the f-th content.

In order to avoid any kind of bottleneck during the delivery of users' content requests, we assume that each SBS has a finite storage capacity of S_m and caches selected contents from the library $\mathcal{F}_{tar}$. Thus, the amount of requests SBSs satisfy from their local caches is of high importance to avoid peak demands and minimize the latency of content delivery. Our goal is to offload the backhaul while satisfying users' content requests, by pre-fetching strategic contents from the CN at suitable times and cache them at the SBSs, subject to their storage constraints. To formalize this, suppose that D number of requests from the set $\mathcal{D} = \{1, ..., D\}$ are made by users during T time-slots. Then, a request $d \in \mathcal{D}$ within time window T is served immediately and is said to be *satisfied*, if the rate of delivery is equal or greater than the content bitrate, such that:

$$\frac{L(f_d)}{\tau'(d) - \tau(d)} \geq B(f_d) \tag{2}$$

where f_d is the requested content, $L(f_d)$ and $B(f_d)$ are the size and bitrate of the content, $\tau(d)$ is the arrival time of the request and $\tau'(d)$ the end time delivery. Given these definitions, the users' average *satisfaction ratio* can be expressed as:

$$\eta(\mathcal{D}) = \frac{1}{D} \sum_{d \in \mathcal{D}} \mathbb{1} \left\{ \frac{L(f_d)}{\tau'(d) - \tau(d)} \geq B(f_d) \right\} \tag{3}$$

where $\mathbb{1}\{...\}$ is the indicator function which returns 1 if the statement holds and 0 otherwise. Suppose that the instantaneous backhaul rate for the content delivery of request d at time t is given by $R_d(t) \leq C_m$, $\forall m \in \mathcal{M}_{tar}$. Then, the average *backhaul load* is defined as:

$$\rho(\mathcal{D}) = \frac{1}{D} \sum_{d \in \mathcal{D}} \frac{1}{L(f_d)} \sum_{t=\tau(f_d)}^{\tau'(f_d)} R_d(t). \tag{4}$$

Now, denote $\mathbf{X}(t) \in \{0,1\}^{M_{tar} \times F_{tar}}$ as the cache decision matrix of SBSs, where $x_{m,f}(t)$ equals 1 if the f-th content is cached at the m-th SBS at time t, and 0 otherwise. Therefore, the backhaul offloading problem can be formally expressed as:

$$\underset{\mathbf{X}(t), \mathbf{P}^m(t)}{\text{minimize}} \quad \rho(\mathcal{D}) \tag{5}$$

$$\text{subject to} \quad L_{\min} \leq L(f_d) \leq L_{\max}, \qquad \forall d \in \mathcal{D},$$
$$B_{\min} \leq B(f_d) \leq B_{\max}, \qquad \forall d \in \mathcal{D},$$
$$R_d(t) \leq C_m, \qquad \forall t, \forall d \in \mathcal{D}, \forall m \in \mathcal{M}_{tar},$$
$$R'_d(t) \leq C'_m, \qquad \forall t, \forall d \in \mathcal{D}, \forall m \in \mathcal{M}_{tar},$$
$$\sum_{f \in \mathcal{F}_{tar}} L(f) x_{m,f}(t) \leq S_m, \quad \forall t, \forall m \in \mathcal{M}_{tar},$$
$$\sum_{n \in \mathcal{N}_{tar}} \sum_{f \in \mathcal{F}_{tar}} P_{n,f}^m(t) = 1, \quad \forall t, \forall m \in \mathcal{M}_{tar},$$
$$x_{m,f}(t) \in \{0,1\}, \quad \forall t, \forall f \in \mathcal{F}_{tar}, \forall m \in \mathcal{M}_{tar},$$
$$\eta_{\min} \leq \eta(\mathcal{D})$$

where $R'_d(t)$ is the instantaneous wireless link rate for request d and $\eta_{\min}$ is the minimum target satisfaction ratio respectively. In order to solve this problem, a joint optimization of the cache decision $\mathbf{X}(t)$ and the content popularity matrix estimation $\mathbf{P}^m(t)$ is needed. Moreover, solving (5) is very challenging due to:

i) limited backhaul and wireless link capacity as well as the limited storage capacity of SBSs,
ii) large number of users with unknown ratings and library size,
iii) SBSs need to track, learn and estimate users' content popularity/rating matrix $\mathbf{P}^m(t)$ for cache decision while dealing with data sparsity.

For simplicity, we drop now the index of the SBSs and assume that the content popularity is stationary during T time slots, thus $\mathbf{P}^m(t)$ is denoted as $\mathbf{P}_{tar}$. Moreover, for sake of exposition, we restrict ourselves to caching policies in which the contents are stored during the peak-off hours, thus $\mathbf{X}(t)$ remains fixed during the content delivery and represented as $\mathbf{X}$. In the following, we examine the source domain which we exploit when dealing with the sparsity of $\mathbf{P}_{tar}$ in the target domain.

B. Source Domain

As advocated in [2], we leverage the existence of a D2D-based social network overlay made of users' interactions within their social communities, referred as the *source domain*

in the sequel. Specifically, this source domain contains the behaviour of users' interactions within their social communities, modelled as a Chinese restaurant process (CRP) [16]. This constitutes the prior information used in the transfer learning procedure.

In the CRP with parameter β, every customer selects an occupied table with a probability proportional to the number of occupants, and selects the next vacant table with probability proportional to β. More precisely, the first customer selects the first table with probability $\frac{\beta}{\beta} = 1$. The second customer selects the first table with probability $\frac{1}{1+\beta}$, and the second table with probability $\frac{\beta}{1+\beta}$. After the second customer selects the second table, the third customer chooses the first table with probability $\frac{1}{2+\beta}$, the second table with probability $\frac{1}{2+\beta}$ and the third table with probability $\frac{\beta}{2+\beta}$. This stochastic Dirichlet process continues until all customers select their seats, defining a distribution over allocation of customers to tables.

In this regard, the content dissemination in the social network is analogous to the table selection in a CRP. If we view this network as a CRP, the contents as the large number of tables, and users as the customers, we can make an analogy between the content dissemination and the CRP. First, suppose that there exist N_{D2D} users in this network. Let $F_{D2D} = F_0 + F_h$ be the total number of contents in which F_h represents the number of contents with viewing histories and F_0 is the number of contents without history. Denote also $\mathbf{Z}_{D2D} \in \{0,1\}^{N_{D2D} \times F_{D2D}}$ as a random binary matrix indicating which contents are selected by each user, where $z_{n,f} = 1$ if the n-th user selects the f-th content and 0 otherwise. Then, it can be shown that [16]:

$$P(\mathbf{Z}_{D2D}) = \frac{\beta^{F_h} \Gamma(\beta)}{\Gamma(\beta + N_{D2D})} \prod_{f=1}^{F_h} (m_f - 1)! \tag{6}$$

where $\Gamma(.)$ is the Gamma function, m_f is the number of users assigned to content f (i.e., viewing history) and F_h is the number of contents with viewing histories with $m_f > 0$.

In the target domain, the caching problem boils down to estimating the content popularity matrix which is assumed to be largely unknown, yielding degraded performance (i.e., very low cache hit ratios, slow convergence, etc.). Moreover, this degradation can be more severe in cases where the number of users and library size is extremely large. Therefore, in order to handle these issues and cache contents more efficiently, we propose a novel proactive caching procedure using transfer learning which exploits the rich contextual information extracted from users' social interactions. This caching procedure is shown to yield more backhaul offloading gains compared to a number of baselines, including random caching and the classical CF-based estimation methods [2].

III. TRANSFER LEARNING: BOOSTING CONTENT POPULARITY MATRIX ESTIMATION

First, we start by explaining the classical CF-based learning, then detail our proposed TL solution.

A. Classical CF-based Learning

The classical CF-based estimation procedure is composed of a training and prediction phase. In the training part, the goal is to estimate the content popularity matrix $\mathbf{P}_{tar} \in \mathbb{R}^{N_{tar} \times F_{tar}}$, where each SBS constructs a model based on the already available information (i.e., users' content ratings). Let $\mathcal{N}_{tar}$ and $\mathcal{F}_{tar}$ represent the set of users and contents associated with N_{tar} users and F_{tar} contents. In particular, $\mathbf{P}_{tar}$ with entries $P_{tar,ij}$ is the (sparse) content popularity matrix in the target domain. $\mathcal{R}_{tar} = \{(i,j,r) : r = P_{tar,ij}, P_{tar,ij} \neq 0\}$ denotes the set of known user ratings. In the prediction phase, in order to predict the unobserved ratings in $\mathcal{N}_{tar}$, low-rank matrix factorization techniques are used to estimate the unknown entries of $\mathbf{P}_{tar}$. The objective here is to construct a k-rank approximate popularity matrix $\mathbf{P}_{tar} \approx \mathbf{N}_{tar}^T \mathbf{F}_{tar}$, where the factor matrices $\mathbf{N}_{tar} \in \mathbb{R}^{k \times N_{tar}}$ and $\mathbf{F}_{tar} \in \mathbb{R}^{k \times F_{tar}}$ are learned by minimizing the following cost function:

$$\underset{(i,j) \in \mathbf{P}_{tar}}{\text{minimize}} \quad \sum_{(i,j) \in \mathbf{P}_{tar}} \left(\mathbf{n}_i^T \mathbf{f}_j - P_{tar,ij} \right)^2 + \quad (7)$$
$$\mu \left(||\mathbf{N}_{tar}||_F^2 + ||\mathbf{F}_{tar}||_F^2 \right)$$

where the sum is over the (i,j) user/content pairs in the training set. In addition, $\mathbf{n}_i$ and $\mathbf{f}_j$ represent the i-th and j-th columns of $\mathbf{N}_{tar}$ and $\mathbf{F}_{tar}$ respectively, and $||.||_F^2$ denotes the Frobenius norm. In (7), the parameter μ provides a balance between regularization and fitting training data. Unfortunately, users may rate very few contents, causing $\mathbf{P}_{tar}$ to be extremely sparse, and thus (7) suffers from severe over-fitting issues and engenders poor performance.

B. TL-based Content Caching

To alleviate data sparsity, solving (7) can be done more efficiently by exploiting and transferring the vast amount of available user-content ratings (i.e., prior information) from a different-yet-related source domain. Formally speaking, let us denote the source domain as $S^{(\mathcal{S})}$, and assume that this domain is associated with a set of N_{D2D} users and F_{D2D} contents denoted by $\mathcal{N}_{D2D}$ and $\mathcal{F}_{D2D}$ respectively. Additionally, the user-content popularity matrix in the source domain is given by matrix $\mathbf{P}_{D2D} \in \mathbb{R}^{N_{D2D} \times F_{D2D}}$ and likewise let $\mathcal{R}_{D2D} = \{(i,j,r) : r = P_{D2D,ij}, P_{D2D,ij} \neq 0\}$ represent the set of observed user ratings in the source domain. The underlying principle of the proposed approach is to smartly "borrow" carefully-chosen user social behavior information from $S^{(\mathcal{S})}$ to better learn $S^{(\mathcal{T})}$.

The transfer learning procedure from $S^{(\mathcal{S})}$ to $S^{(\mathcal{T})}$ is composed of two interrelated phases. In the first phase, a content *correspondence* is established in order to identify similarly-rated contents in both source and target domains. In the second phase, an optimization problem is formulated by combining the source and target domains for *knowledge transfer*, to jointly learn the popularity matrix $\mathbf{P}_{tar}$ in the target domain. In this regard, we suppose that both source and target domains correspond to one information system $s \in \{S^{(\mathcal{S})}, S^{(\mathcal{T})}\}$, that is made of N_s users and F_s contents given by $\mathcal{N}_s$

and $\mathcal{F}_s$ respectively. In each system s, we observe $\mathbf{P}_s$ with entries $P_{s,ij}$. Let $\mathcal{R}_s = \{(i,j,r) : r = P_{s,ij}, P_{s,ij} \neq 0\}$ represent the set of observed user ratings in each system and the set of *shared contents* is given by $\tilde{\mathcal{F}}$. Moreover, let $\mathcal{N}^* = \mathcal{N}_{D2D} \cup \mathcal{N}_{tar}$ and $\mathcal{F}^* = \mathcal{F}_{D2D} \cup \mathcal{F}_{tar}$ be the union of the collections of users and contents, respectively, where $N^* = |\mathcal{N}^*|$ and $F^* = |\mathcal{F}^*|$ represent the total number of unique users and contents in the union of both systems.

In the proposed TL approach, we model the users $\mathcal{N}^*$ and contents $\mathcal{F}^*$ by a user factor matrix $\mathbf{N} \in \mathbb{R}^{k \times N^*}$ and a content factor matrix $\mathbf{F} \in \mathbb{R}^{k \times F^*}$, where the i-th and j-th columns of these matrices are given by $\mathbf{n}_i$ and $\mathbf{f}_j$, respectively. The aim is to approximate the popularity matrix $\mathbf{P}_s \approx \mathbf{N}_s^T \mathbf{F}_s$ by jointly learning the factor matrices $\mathbf{N}$ and $\mathbf{F}$. This is formally done by minimizing the following cost function:

$$\underset{(i,j) \in \mathbf{P}_s}{\text{minimize}} \quad \sum_s \left(\alpha_s \sum_{(i,j) \in \mathbf{P}_s} \left(\mathbf{n}_i^T \mathbf{f}_j - P_{s,ij} \right)^2 \right) + \quad (8)$$
$$\mu \left(||\mathbf{N}||_F^2 + ||\mathbf{F}||_F^2 \right)$$

where the parameter α_s is the weight of each system. By doing so, $\mathbf{P}_{D2D}$ and $\mathbf{P}_{tar}$ are jointly factorized, and thus the set of factor matrices $\mathbf{F}_{D2D}$ and $\mathbf{F}_{tar}$ become interdependent as the features of a shared content are similar for knowledge sharing. A practical TL-based caching procedure is sketched in Fig. 2.
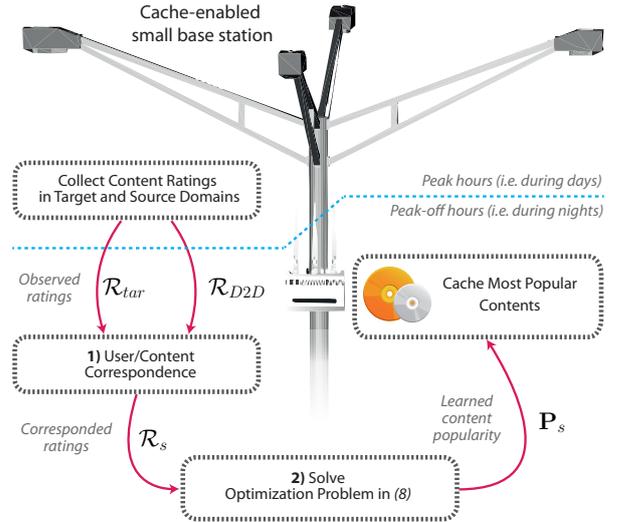

Figure 2: An illustration of the proposed TL-based caching procedure.

IV. NUMERICAL RESULTS AND DISCUSSION

The objective of this section is to validate the effectiveness of the proposed TL caching procedure and draw key insights. In particular, we consider the following caching policies for comparison:

1) *Ground Truth*: Given the perfect rating matrix $\mathbf{P}_{tar}$, the most popular contents are stored greedily.
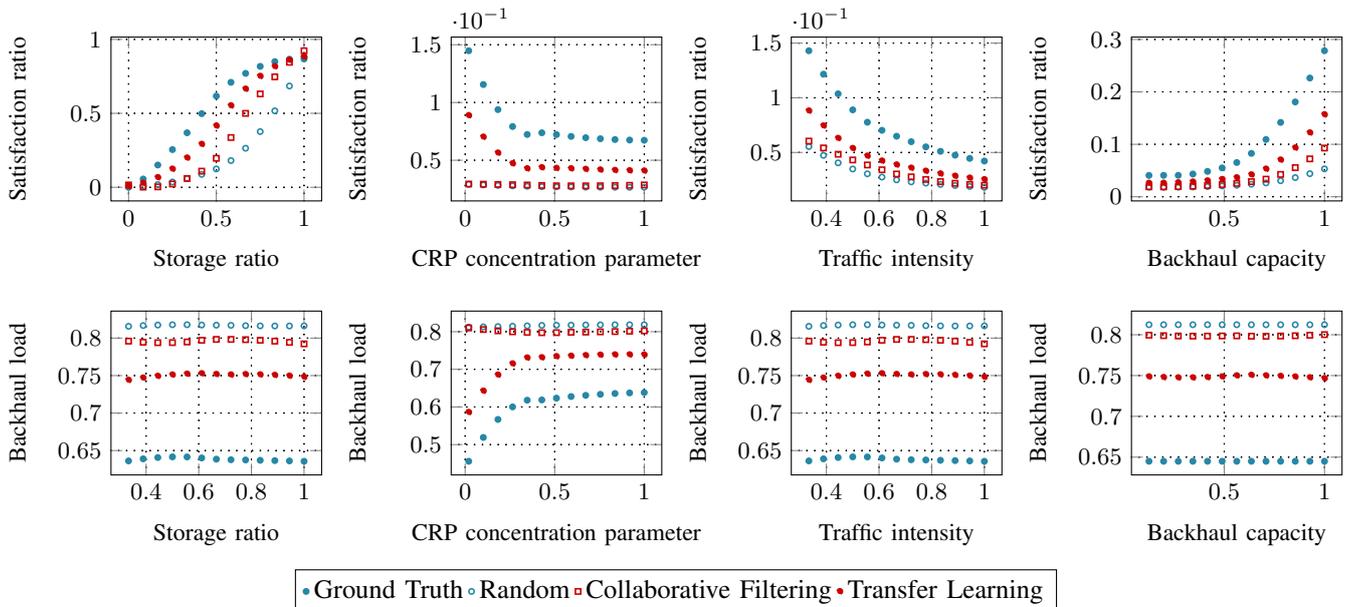2) *Random caching* [2]: Contents are cached uniformly at random.

Figure 3: Evolution of the aggregate backhaul load and users' satisfaction ratio.

3) *Collaborative Filtering* [13]: The content popularity matrix $\mathbf{P}_{tar}$ is estimated via CF from a training set with 4% of ratings. Then, the most popular contents are stored accordingly.

4) *Transfer Learning*: $\mathbf{P}_{tar}$ and $\mathbf{P}_{D2D}$ matrices are jointly factorized via TL by using a training set with 12% of ratings and perfect user-content correspondence. Then the most popular contents are stored accordingly.

In the numerical setup, having contents cached according to these policies, the SBSs serve their users according to a traffic arrival process. This process is drawn from a Poisson process with intensity λ. The storage size of SBSs, content lengths, capacities of non-interfering wireless and backhaul links are assumed to have same constant values individually, in order to showcase the performance of the caching policies. The numerical results of users' satisfaction ratio and backhaul load are obtained by averaging out 1000 Monte-Carlo realizations. The simulation parameters are summarized in Table I, unless stated otherwise.

The dynamics of users' satisfaction ratio and backhaul load with respect to the storage size, demand shape in the source domain, traffic intensity and backhaul capacity are given in Fig. 3. The results are normalized to show the various percentage gains, whereas the actual values are shown in Table I. In the following, we discuss in detail the impact of these parameters.

1) Impact of the storage size (S_m): The storage size is indeed one of the crucial parameter in cache-enabled SBSs, and it is expected that higher storage sizes result in better performance in terms of satisfaction ratio and backhaul offloading. According to this setup, we would like to note that the biggest improvement in satisfaction ratio and decrement in the backhaul load is achieved by the ground truth baseline

Table I: Simulation Parameters

Parameter	Description	Default-Varied Values
M_{tar}	Number of SBSs	4
N_{tar}	Number of UTs	32
F_{tar}	Library size	32 contents
L	Content length	1 MBit
B	Bitrate requirement	1 MBit
$\sum C'_m$	Total wireless capacity	32 MBit/s
T	Time slots	128 seconds
α	Zipf parameter	2
β	CRP concentration parameter	2 - $[2 \sim 100]$
$\sum S_m$	Total storage size	6 - $[0 \sim 32]$ MBit
$\sum C_m$	Total backhaul capacity	1 - $[1 \sim 8]$ MBit/s
λ	Traffic intensity	1 - $[1 \sim 3]$ demand/s

where the content popularity is perfectly known. The random approach on the other hand has the worst-case performance. The CF approach exhibits similar performance as the random approach due to the cold-start problem, whereas the satisfaction ratio and backhaul offloading gains of TL are close to the ground truth baseline. In particular, it is shown that the TL policy outperforms its CF counterpart, with satisfaction and backhaul offloading gains up to 22% and 5% respectively.

2) Impact of the demand shape in the source domain (β): The demand shape in the source domain, characterized by the CRP concentration parameter β provides meaningful insights to our problem. In fact, as β increases, the demand shape tends to be more uniform, requiring higher storage sizes at the SBSs to sustain the same performance. In a storage limited case, we see that the satisfaction ratio decreases and the backhaul load increases with the increment of β. Compared to the CF

approach, the gains of TL are around 6% for the satisfaction gains and 22% for the backhaul offloading. However, the gap between TL and CF becomes smaller as β increases.

3) Impact of the traffic intensity (λ): As the average number of request arrivals per time slot increases, bottlenecks in the network are expected to occur due to the limited resources of SBSs, resulting in less satisfaction ratios. This is visible in the high arrival rate regime, whereas the relative backhaul load remains constant. It can be shown that the ground truth caching with perfect knowledge of content popularity outperforms the other policies while the random approach has the worst performance. On the other hand, the performance of TL is in between these approaches and has up to 3% satisfaction gains and 18% of backhaul offloading gain compared to the CF.

4) Impact of the backhaul capacity (C_m): The total backhaul capacity is assumed to be sufficiently smaller than the capacity of wireless links. The increment of this capacity clearly results in higher satisfaction ratios in all cases. Note that any content not available in the caches of SBSs is delivered via the backhaul. Therefore, increasing the backhaul capacity avoids the bottlenecks during the delivery, thus yielding higher users' satisfaction. On the other hand, the backhaul load remains constant in this setting. It can be seen that TL approach has satisfaction ratio gains of up to 6% and backhaul offloading of up to 5% compared to the CF approach.
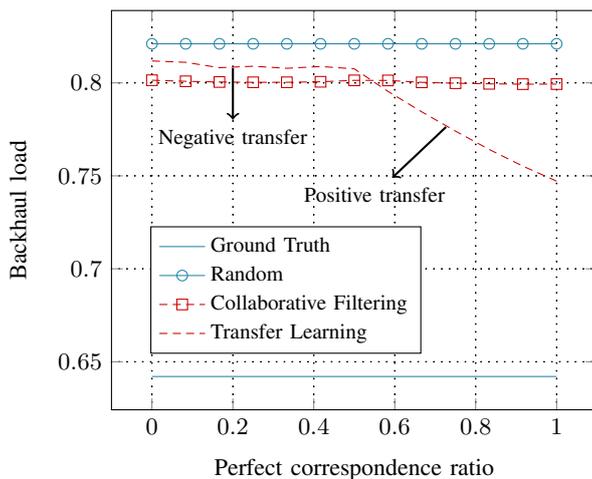

Figure 4: Evolution of the backhaul load with respect to the perfect correspondence ratio.

5) Impact of source-target correspondence: We have so far assumed that the user/content correspondence between the target and source domains is perfect. This is a strong assumption and such an operation requires a more careful treatment to avoid negative transfer. Here, we relax this assumption by introducing a perfect correspondence ratio. This ratio represents the amount of perfect user/content matching between both source and target domains. A ratio of 0 means that 100% of correspondence is done uniformly at random and 1 is equivalent to the perfect case. It is shown in Fig. 4 that TL has a poor performance in the low values of this

ratio, with similar performance as the random caching due to the negative transfer. However, as this ratio increases, the performance of TL improves, outperforming the CF with a ratio of 0.58. This underscores the importance of such an operation for the positive transfer and is left for future work.

V. Conclusions

We proposed a novel transfer learning-based caching procedure which was shown to yield higher users' satisfaction and backhaul offloading gains overcoming the data sparsity and cold start problems. Numerical results confirmed that the overall performance can be improved by transferring a judiciously-extracted knowledge from a source domain to a target domain via TL. An interesting future work is assessing the performance of TL-based caching using real traces. Another avenue of research is extending the current model to predictive scheduling and predictive offloading.

References

[1] F. Boccardi, R. Heath, A. Lozano, T. Marzetta, and P. Popovski, "Five disruptive technology directions for 5g," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, February 2014.

[2] E. Baştuğ, M. Bennis, and M. Debbah, "Living on the Edge: The role of proactive caching in 5G wireless networks," *IEEE Communications Magazine*, vol. 52, no. 8, pp. 82–89, August 2014.

[3] N. Golrezaei, K. Shanmugam, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless video content delivery through distributed caching helpers," in *IEEE INFOCOM*, March 2012, pp. 1107–1115.

[4] E. Baştuğ, M. Bennis, M. Kountouris, and M. Debbah, "Cache-enabled small cell networks: Modeling and tradeoffs," *EURASIP Journal on Wireless Communications and Networking*, Accepted (2014).

[5] A. Altieri, P. Piantanida, L. R. Vega, and C. Galarza, "On fundamental trade-offs of device-to-device communications in large wireless networks," *arXiv preprint arXiv:1405.2295*, 2014.

[6] P. Blasco and D. Gunduz, "Learning-based optimization of cache content in a small cell base station," *arXiv preprint arXiv:1402.3247*, 2014.

[7] K. Poularakis, G. Iosifidis, and L. Tassiulas, "Approximation algorithms for mobile data caching in small cell networks," *to appear in IEEE Transactions on Communications*, 2014.

[8] K. Hamidouche, W. Saad, and M. Debbah, "Many-to-many matching games for proactive social-caching in wireless small cell networks," in *WNC3 workshop, WiOpt*, Hammamet, Tunisia, May 2014.

[9] M. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Transactions on Information Theory*, vol. 60, no. 5, pp. 2856–2867, May 2014.

[10] J. Pääkkönen, C. Hollanti, and O. Tirkkonen, "Device-to-device data storage for mobile cellular systems," in *IEEE Globecom Workshops (GC Wrokshops)*, December 2013, pp. 671–676.

[11] M. S. ElBamby, M. Bennis, W. Saad, and M. Latva-aho, "Content-aware user clustering and caching in wireless small cell networks," in *11th International Symposium on Wireless Communication Systems (ISWCS)*, Barcelona, Spain, August 2014.

[12] A. Liu and V. Lau, "Cache-enabled opportunistic cooperative mimo for video streaming in wireless systems," *IEEE Transactions on Signal Processing*, vol. 62, no. 2, pp. 390–402, January 2014.

[13] J. Lee, M. Sun, and G. Lebanon, "A comparative study of collaborative filtering algorithms," *[Online] arXiv: 1205.3193*, 2012.

[14] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, October 2010.

[15] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and zipf-like distributions: Evidence and implications," in *IEEE Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'99)*, vol. 1. IEEE, 1999, pp. 126–134.

[16] T. L. Griffiths and Z. Ghahramani, "The Indian Buffet Process: An Introduction and Review," *J. Mach. Learn. Res.*, vol. 12, pp. 1185–1224, July 2011.