Signal Processing: *Image Communication* 29 (2014) 462–471

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

journal homepage: www.elsevier.com/locate/image

Social video caching

Jacob Chakareski

Department of Electrical and Computer Engineering, The University of Alabama, Tuscaloosa, AL 35487, The United States

ARTICLE INFO

Article history:
Received 5 May 2013
Received in revised form
6 February 2014
Accepted 24 February 2014
Available online 5 March 2014

Keywords:
Social video
Caching
Content delivery systems

ABSTRACT

We investigate online browsing of interrelated content, represented as a catalog of items of interest featuring graph dependencies. The content is served to clients via a system of decentralized proxy caches connected to cloud servers. A client selects the next item to browse from the list of recommended items, displayed on the currently browsed item's catalog page. A cache has a limited size to have every item selected by its browsing clients available for local access. Thus, the system pays a penalty, whenever a client selects an item that cannot be served directly from the proxy. Conversely, the system gains a reward, if a client selects an immediately available item. We aim to select the items to cache that maximize the profit earned by the system, for the given cache capacity. We design two linear-time optimization techniques for finding the desired items to cache. We enhance the operation of the system via two additional strategies. The first one dynamically tracks the items' selection probabilities for a client, as a function of its prior catalog access pattern and those of its community peers. The second one constructs dynamic overlays, on behalf of the clients, that are used to share the selected items directly among them. This augments the system's serving capacity and enhances the clients' browsing experience. We study the performance of the optimization techniques via numerical experiments. They exhibit efficiency gains over reference methods, by exploiting the content dependencies and correlated community-driven access patterns of the clients. We also report proxy bandwidth savings achieved by our overlay strategy over state-of-the-art methods, on content access patterns of clients with Facebook or Twitter ties.

1. Introduction

Frequently, we encounter the scenario of browsing content items in a large catalog by hopping from one item to another. For example, we browse the web by clicking on the URL links embedded on each page we visit. Similarly, we shop for items of interest online by checking the recommended products that appear together with every item we examine. Finally, watching videos on YouTube by repeatedly clicking on one of the links to related videos, for every clip we just watched, represents yet another example of this very general setup. In each case, there is a reward to be gained by providing immediate access to the next selected item. In particular, the user will be more satisfied with the service and therefore will be more motivated to continue browsing the catalog. These in turn can result in extended customer loyalty, longer browsing times, and ultimately higher profit for the service/content provider. On the other hand, there is a penalty that the provider will need to bear, if the user selects an item in the catalog that is not immediately available. That is because the provider will need to deliver this item from a remote center and support the related cost.[1] At the same time, the user may lose interest in further browsing the catalog while waiting for the currently selected item to be fetched, which equally can manifest itself

E-mail address: jakov@jakov.org

[1] Transiting traffic across different ISP networks is always subject to cost.

http://dx.doi.org/10.1016/j.image.2014.02.002

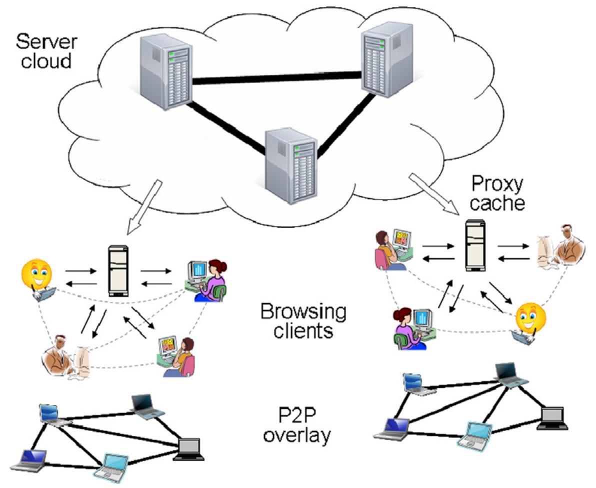

Fig. 1. Distributed content delivery system for community-aware browsing. The content catalog is served from the cloud servers via local proxy caches to browsing clients. Gray dashed lines denote social ties between users. Clients cache content and organize into P2P overlays for enhanced performance.

as a cost (seen as lost profit) to the provider. Due to reasons of complexity and limited resources, the service provider can only maintain a subset of the content items available for direct access by its users. They are typically delivered via a proxy cache located within the same ISP network as the users. The complete system features a collection of decentralized caches, spanning multiple ISPs and connected to cloud servers, as illustrated in Fig. 1. Any content item unavailable locally is served from the cloud.

In such a setup, the service provider is interested in selecting the subset of catalog items that should be immediately available from a proxy cache such that its profit is maximized, given the interdependencies between the content items introduced by the catalog, the capacity of the cache, and the clients' catalog access patterns. We formulate the reward that the provider earns as an expectation over the set of prospective actions of the clients on the catalog graph, over a horizon of time. We then solve the problem of interest as a constrained optimization of the expected reward over all prospective content subsets that do not exceed the cache's size. The optimization tracks dynamically the users' content access patterns and their community ties to account for their impact on the catalog's caching efficiency. The overall performance of the system is further enhanced by integrating a peer-to-peer (P2P) strategy that allows the users to share catalog items via overlay networks, as illustrated in Fig. 1.

The problem we investigate is important, due to the immense popularity of social multimedia sharing applications that operate on extensively large data sets that exhibit graph dependencies. For instance, the YouTube video library already accounts for at least ten petabytes of data, even by very conservative estimates, due to its extraordinary growth rate of 65,000 video clips per day. Such a high data volume can only be stored in distributed databases operated from remote data centers sited at different locations. According to recent statistics, there are more than 2 billion page views served by YouTube every day [1]. A traffic request volume of such a magnitude will inevitably lead to delivering a major part of the respective video content remotely. Thus, in addition to prospectively losing audience as the content is not immediately accessible, the provider also faces a non-negligible data transport operating cost caused by the sheer scale of this traffic. Challenges of the same nature are encountered by other leading providers of online services such as Amazon. It is expected that highly interactive media-rich and data intensive applications will increasingly dominate the Internet [2], thereby emphasizing even further the very current problem of efficient large-scale data management online.

The rest of the paper is organized as follows. Our system models are characterized in Section 2. Then, we formulate the problem under consideration in Section 3. Showing that the resulting objective function is monotonic and sub-modular is carried out in Section 4. As a result of special properties, we can derive in Section 5 the two low-complexity linear-time techniques for solving the optimization problem. A brief complexity analysis of the techniques is provided in Section 6. We study their performance via experimentation in Section 7 that includes an evaluation of the networking efficiency of the P2P technique that is integrated within the system. We discuss related work in detail in Section 8. Finally, we conclude in Section 9.

2. System model

2.1. Content browsing

Each page of the catalog features a content item that a client can select to browse and a list of recommended related items that is linked to their respective pages in the catalog. We consider that a client can select any item in the catalog $\mathcal{C}$ initially. The probability that client k selects item $i \in \mathcal{C}$ is denoted as $\pi_k(i)$. Each subsequent selection is then limited to the list of recommended items on that page. In particular, let i denote the present item that the client is browsing and let $\mathcal{N}_i$ denote the recommended items' set. The client can choose the next item $j \in \mathcal{N}_i$ to sample with probability p_{ij}, where $\sum_j p_{ij} = 1$. Our behavioral model is based on empirical studies reported in the context of web article browsing, online shopping, and content sharing sites. Section 2.4 provides further characterization of this model, as considered here.

The subset of catalog items available for local access is denoted as $\mathcal{A}$. Similarly, $\overline{\mathcal{A}}$ denotes the set of content items that the proxy must deliver remotely. Given that $\mathcal{C}$ represents the set of items comprising the whole catalog, it holds $\overline{\mathcal{A}} = \mathcal{C} \backslash \mathcal{A}$. Next, let A_q represent the reward (profit) that the system will earn by serving item q from the cache. Conversely, let B_q denote the penalty (cost) that the system will bear, when a client selects an item $q \in \overline{\mathcal{A}}$, as illustrated in Fig. 2. Finally, let S denote the capacity of the proxy cache.

2.2. Social relationships

We assume that a client can prospectively belong to an online community, formally represented by the triple $G = (V, E, W)$, where G denotes the social graph that characterizes