Journal of Network and Computer Applications 37 (2014) 273–281

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

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

A novel cache size optimization scheme based on manifold learning in Content Centric Networking

Yuemei Xu [a,*], Yang Li [a], Tao Lin [a], Zihou Wang [a], Wenjia Niu [b], Hui Tang [a], Song Ci [a,c]

[a] High Performance Network Lab, Institute of Acoustics, Chinese Academy of Sciences, Beijing 100190, China
[b] Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100190, China
[c] Department of Computer and Electronics Engineering, University of Nebraska-Lincoln, NE 68182, USA

ARTICLE INFO

Article history:
Received 29 August 2012
Received in revised form
1 March 2013
Accepted 2 March 2013
Available online 13 March 2013

Keywords:
Content Centric Networking
Cache size allocation
Manifold learning

ABSTRACT

Content Centric Networking (CCN) is an emerging network architecture, shifting from an *end-to-end* connection to a *content centric* communication model. Each router in CCN has a content store module to cache the chunks passed by, and is arranged in an arbitrary network topology. It is important to allocate an appropriate cache size to each router in order to both improve the network performance and reduce the economic investment. Previous works have proposed several heterogeneous cache allocation schemes, but the gain brought by these schemes is not obvious. In this paper, we introduce a data mining method into the cache size allocation. The proposed algorithm uses manifold learning to analyze the regularity of network traffic and user behaviors, and classify routers based on their roles in the content delivery. Guided by the manifold learning embedding results, a novel cache size optimization scheme is developed. Extensive experiments have been performed to evaluate the proposed scheme. Simulation results show that the proposed scheme outperforms the existing cache allocation schemes in CCN.

1. Introduction

Content Centric Networking (CCN) is a novel networking paradigm centered around content distribution rather than host-to-host connectivity (Carofiglio et al., 2011b, 2011a). This change from *host-centric* to *content centric* has several attractive advantages, such as network load reduction, low dissemination latency, etc. Each router in CCN is equipped with a content store (CS) module to cache the data passed by. Users' requests routed towards the original server may be satisfied by an intermediate router with a temporary data copy.

Caching has been extensively studied in the past decades (Breslau et al., 1999; Zhu et al., 2010). However, a number of *content-oriented* characteristics make caching in CCN different from the previous works. Firstly, caching becomes an *intrinsic* property of CCN routers. Both content caching and request routing in CCN are operated at a same network layer, which requires the content retrieval and replacement to operate at *line speed* (Rossi and Rossini, 2011). Specifically, before a router sends a request to the next hop, it should check whether its local CS has the requested content or not. The memory access speed limits the real cache size. Secondly, the ubiquitous caches in CCN are arranged as an *arbitrary* graph topology rather than a *hierarchical* topology with limited nodes as that in Web-caching or in Content Delivery Network (CDN). Thirdly, multiple applications can use the same cache space in CCN since content chunks are identified by unique names. This character makes caching in CCN fundamentally different from the traditional caching system, like Web, CDN and P2P.

In response to these characteristics of CCN, many researches recently fall into the in-network caching study in CCN, including the cache placement, cache replacement and network cache model, etc. In this paper, we are concerned with the cache size allocation, which is the basis of other cache related researches. *Line speed requirements* and the *arbitrary graph topology* are the two features to investigate the cache size allocation. From the perspective of system performance, routers equipped with high-speed hardware are more possible to satisfy the *line speed requirements* in supporting the name-based data forwarding and caching. For example, DRAM (Dynamic Random-Access Memories) can support 10 Gbps line speed, since its access time is about 55 ns, 200 times faster than hard disk drives (Perino and Varvello, 2011). From the perspective of economic investment, the high-speed hardware costs much more than the hard disks. In the *arbitrary graph topology*, all distinct pairs of nodes have inexplicit upstream or downstream relationships. The deployment of nodes are distributed, ubiquitous, and high dense. It is not rational to

* Corresponding author. Tel.: +86 10 15110099943.
E-mail addresses: xuym@hpnl.ac.cn (Y. Xu), liy@hpnl.ac.cn (Y. Li), lint@hpnl.ac.cn (T. Lin), wangzh@hpnl.ac.cn (Z. Wang), nwj6688@gmail.com (W. Niu), tangh@hpnl.ac.cn (H. Tang), sci@engr.unl.edu (S. Ci).

1084-8045/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jnca.2013.03.002

allocate every router with the same cache size. An intuitive idea is to allocate relative larger cache sizes to some important nodes and smaller sizes to the rest regular nodes. However, it is challenging to distinguish nodes' importance and few works focus on the judgement of caching nodes' importance in CCN.

Existing work has exploited the network topology information to distinguish routers and allocate heterogeneous cache size to routers in CCN (Rossi and Rossini, 2012). However, the profit brought by this graph-related approach is not obvious. We regard that the nodes' importance should be determined by their roles in content delivery instead of their topology locations. This means that given the same cache space, the nodes which contribute greater to the content distribution should be considered more important. The determination of node importance involves many interrelated factors. Instead of the network topology information, the factors that reflect the situation of content delivery are more important, such as network traffic distributions and user behaviors. Both of these two factors may fluctuate in a short period of time, but are relatively stable from a long-term observation, which is called *locality characteristic* (Rossini and Rossi, 2011).

In this paper, we introduce a data mining method to characterize the network traffic distributions and user behaviors, in order to figure out which nodes are important. We collect multidimensional information as the raw mining data and use the data mining results to distinguish each node's importance. Then we cluster nodes according to their importance and assign different cache sizes to different categories of nodes. The major contributions of this paper are summarized as follows:

- Distinguish the degree of nodes' importance through analyzing the network traffic distributions and user behaviors. The proposed cache size allocation scheme is based on nodes' importance property.
- Introduce a nonlinear manifold learning method to mine the collected multidimensional raw data and execute the node clustering.
- Evaluate the proposed cache size allocation scheme on a chunk-level CCN simulation environment. The results show that our approach greatly improves the network performance comparing with other schemes.

The rest of the paper is organized as follows. Section 2 presents a survey of the related work and the background of manifold learning method. Section 4 illustrates the manifold-learning based algorithm and the corresponding cache size allocation scheme is shown in Section 5. Section 6 is the simulation evaluation. Finally, conclusions are summarized in Section 7.

2. Related work and background

2.1. Related work

The problem of cache size allocation has been investigated in the context of memory caches, on-line Web caching and CDN system. For example, Kelly and Reeves (2001) focus on the problem of determining exact optimal storage capacity for Web caches. It exclusively concerns on the economic considerations, and gives exact solution for the formulated problem. Laoutaris et al. (2005) consider the problem of how to best allocate an available storage budget to the nodes in a hierarchical content distribution system. Furthermore, Laoutaris et al. (2005) jointly consider the related resource allocation factors (i.e., node locations, object placement), and model the optimization problem as an integer linear program for the first time. However, these existing works are based on a hierarchical network architecture and consider cache size allocation only in a few of proxy servers. The contents caching and requests routing in traditional Web caches are operated at different network layers without line speed requirements. While in the context of CCN, all the caching nodes are arranged in an arbitrary topology and have to obey the line speed requirements. The aforementioned solutions no longer fit for CCN cache size allocation.

Efforts (Rossi and Rossini, 2012) have been made to explore an appropriate cache size for an individual CCN router. It is inspired by the heterogeneous replacement policy in Web caches, and tries to investigate a *heterogeneous cache size* allocation by exploiting the network topological information. Rossi and Rossini (2012) propose to allocate cache size according to some graph-related centrality properties (e.g., betweenness, closeness, stress, graph, eccentricity and degree centralities). For example, using *degree centrality* criterion, nodes with larger number of connected links are considered more important, so as to be assigned with more cache space. Nodes are assigned with different cache sizes proportional to their degree values. However, disappointingly the conclusion given by Rossi and Rossini is that the performance of these graph-related schemes is more or less as same as the most simplified homogeneous allocation scheme. The limitation of the graph-related schemes lies in that they do not consider the network traffic distributions and user behaviors, which we regard as the crucial factors to reflect nodes' importance. In this paper, we propose a novel cache size allocation scheme by considering the influence of network traffic distributions and user behaviors. We use a data mining method to find the regularity of these two factors.

We adopt a nonlinear manifold learning method to perform the data mining and clustering. There are many learning-based clustering algorithms, such as K-Means (Jain, 2010), CURE (clustering using representative) (Patid et al., 2012), etc. However, most of the learning-based clustering algorithms need to know some priori information (e.g., the number of clustering, locations of nodes), and rely heavily on the algorithms' initial values. In this paper, the raw data comes from the network measurement variables, which is multidimensional and nonlinear. It is unable to get the priori knowledge of cache size allocation. For example, the appropriate number of clustering is not available at the beginning, which can be obtained only after getting the regularity of the collected data. The nonlinear manifold learning method has been widely used for dimensionality reduction without explicitly setting the initial values of algorithms or knowing the network information, and can directly show the relationships of the multidimensional data in a lower dimension (Law and Jain, 2006).

2.2. Nonlinear manifold learning

Nonlinear dimensionality reduction (NLDR) is an effective method to discover meaningful low-dimensional structures hidden in high-dimensional data. Often, the high-dimensional data observed in the real world are the consequences of a small number of factors. Manifold learning is one of the approaches for dimensionality reduction. Specially, let X be D-dimensional observed data in $\mathbf{R}^D$, and Y be a d-dimensional domain contained in the Euclidean space $\mathbf{R}^d$, normally, $d \ll D$. The $f: X \rightarrow Y$ is a smooth embedding. The object of manifold learning is to discover a low dimensional data Y and a embedding f based on a given observed high-dimensional data X in $\mathbf{R}^D$. Figure 1 depicts an example of dimensionality reduction by manifold learning. The Swiss roll on the left of Fig. 1 is a 3-dimensional distribution space, while the dots on the right of Fig. 1 are the *intrinsic* representations of Swiss roll in a 2-dimensional plane.