Available online at www.sciencedirect.com

ScienceDirect

December 2017, 24(6): 1–13
www.sciencedirect.com/science/journal/10058885

The Journal of China Universities of Posts and Telecommunications

http://jcupt.bupt.edu.cn

Measuring web page complexity by analyzing TCP flows and HTTP headers

Cheng Weiqing[1, 2] (✉), Hu Yangyang[1], Yin Qiaofeng[3], Chen Jiajia[1]

1. School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
2. Key Laboratory of Computer Network and Information Integration, Southeast University, Nanjing 211189, China
3. Jiangsu Education Information Center, Jiangsu Provincial Department of Education, Nanjing 210013, China

Abstract

To understand website complexity deeply, a web page complexity measurement system is developed. The system measures the complexity of a web page at two levels: transport-level and content-level, using a packet trace-based approach rather than server or client logs. Packet traces surpass others in the amount of information contained. Quantitative analyses show that different categories of web pages have different complexity characteristics. Experimental results show that a news web page usually loads much more elements at more accessing levels from much more web servers within diverse administrative domains over much more concurrent transmission control protocol (TCP) flows. About more than half of education pages each only involve a few logical servers, where most of elements of a web page are fetched only from one or two logical servers. The number of content types for web game traffic after login is usually least. The system can help web page designers to design more efficient web pages, and help researchers or Internet users to know communication details.

Keywords hyper text transfer protocol, concurrent TCP flows, world wide web, web page complexity

1 Introduction

Since the world wide web (WWW) was born, it has remained to be the most popular application on the Internet. Over the last twenty years, web pages have become more and more complex, evolving from static texts, to texts with some images, and to rich media mixed with texts, images, graphics, animations, audio, video, flash movies, JavaScript scripts executed on web browsers, and so on. Furthermore, the contents for a web page on a website are fetched not only from servers hosted by the website, but also from servers hosted by online shopping sites, analytics platforms, game sites, content distribution networks (CDN), social platforms, and so on.

The more complex a web page is, the more information it usually contains. But, does the increase in web page complexity imply a better user experience? The answer is definitely not. There are plenty of evidences that the increase in web page complexity is a key factor in slowing down websites [1–2]. Besides, complex pages usually dazzle people, and look irritating. Thus, it is meaningful to study the complexity of web pages, in order to improve web browser design or web page design.

There are many related efforts to analyze web traffic. A packet trace-based approach was used to construct an empirical model of WWW generated network traffic, which consists of a number of probability distributions determined by analysis of actual hyper text transfer protocol (HTTP) conversations [3]. To address the evolution of the web, a light-weight methodology based on passive tracing of only transmission control protocol/ Internet protocol (TCP/IP) headers from one direction (from web servers to web clients) of the TCP connection was used to model web traffic [4]. The analysis results of Ref. [4] held numerous insights into the evolution of the

Received date: 13-03-2017
Corresponding author: Cheng Weiqing, E-mail: chengweiq@njupt.edu.cn
DOI: 10.1016/S1005-8885(17)60237-1

HTTP and the web content between 1995 and 2003. Callahan et al. [5] employed web traffic logs from their intrusion detection system collected over three and a half years to study various aspects of the web, characterize client HTTP transactions, and develop a view of the structure of the web, including an initial understanding of the behavior of browser caches and the impact of CDNs. Newton et al. [6] used methods similar to Ref. [4] to reveal more current trends in web traffic evolution without access to HTTP headers. In Refs. [1–2], the firebug extension add-ons was used to automatically export a log of all the requests and responses involved in rendering a web page. The log is in the HTTP archive record (HAR) format and provides a detailed record of the actions performed by the browser in loading the page. Butkiewicz et al. [1–2] quantified the complexity of a web page with a broad spectrum of metrics and identified the critical complexity metrics that have the most impact on the time to download and render a web page.

We make a deep study of measuring the complexity of a web page using a packet trace-based approach. This work uses packet traces rather than server logs or client logs that are popular in web measurements, because packet traces are easily obtained and more information can be derived from them.

This paper differs from work of Refs. [1–2] on data and metrics used to characterize web page complexity. Packet traces used in the paper differ from those used in Refs. [4,6] in that traces contain raw packets from which not only TCP/IP headers but also HTTP headers can be extracted.

The rest of the paper is organized as: in Sect. 2, the architecture of the web page complexity measurement system is provided, data structures in relation to packet parse and web page complexity analysis are given. In Sect. 3, transport-level metrics and content-level metrics used in the system as well as how to measure them are described. In Sect. 4, experimental results and summary are shown. Finally, in Sect. 5, conclusions are drawn and directions for future research are proposed.

2 Architecture

To measure web page complexity, we design two programs: one is a packet capturer using windows packet capture (WinPcap) on windows, the other is the web page complexity parser, as shown in Fig. 1.

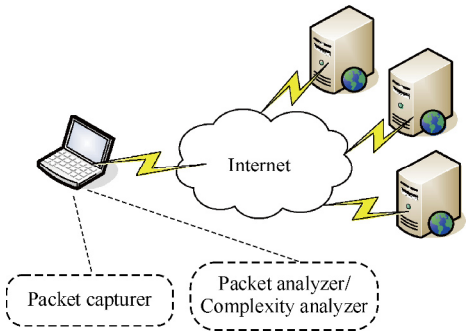

Fig. 1 The architecture of the measurement system

2.1 Packet capturer

To avoid telling apart network traffic for several web pages, we artificially ensure that one packet trace records network traffic for a web page that is caused by a click on a hyperlink or a submitting of a uniform resource locator (URL) in a web browser's address bar. A packet trace contains the Internet protocol (IP) address and the subnet mask of the capturing host, and records of packets. Each record of a packet consists of the time stamp that indicates when this packet is captured, length of this packet (frame), and raw data of this packet.

2.2 Web page complexity parser

The web page complexity parser takes a packet trace as input, and logically consists of two modules: a packet analyzer and a complexity analyzer, shown in Fig. 1.

2.2.1 Packet analyzer

The packet analyzer keeps track of network activities of all local hosts visible to the capturing host. A host is represented by a host structure (Host) shown in Fig. 2.

The linked list of all hosts is accessed through the pointer Hosts. A host structure maintains the local IP address, a doubly linked list of TCP flows if a TCP connection exists, and the number of concurrent TCP flows. A TCP flow structure (TCPFlow) contains flow keys, flow state, flow statistical information, a doubly linked list of TCP segments (TCPSeg), and a doubly linked list of HTTP request-response pairs (HTTPReqResp).

A TCP flow is also defined by 5-tuple (local IP, local port, remote IP, remote port, TCP) [7–8], and a flow is considered to be expired if no more packets belonging to the flow have been observed for a certain period of time. IP packets that shuttle between specific local endpoint