Monitoring and diagnosis of Internet Quality of Experience (QoE)

Résumé du projet de recherche (Langue 1)

As our lives become more dependent on the Internet, users are increasingly frustrated when poor Internet performance prevents them from accomplishing ever-more important online activities. The Quality of Experience (QoE) when accessing the Internet is thus a key factor for today’s society. Poor Internet QoE is frustrating, in particular for most Internet users who are not tech savvy and hence cannot diagnose—let alone fix—problems by themselves. Internet services and their interactions with the networks they use have become increasingly difficult to predict, which hampers the attempts to diagnose QoE, e.g., due to proliferation of proxies and caches at the network core and of home wireless and 3G/4G access. It is hard even for networking experts to fully diagnose and fix problems. When users get frustrated with degraded QoE, they may stop using services, and companies get the blame (and possibly lose money). Despite the relevance of QoE, we still face limitations to correctly predict and assess QoE, especially in real operational deployments, and diagnosing the causes for poor QoE related to the network and application-layer decisions remains an open challenge. First, going from typical Quality of Service (QoS) metrics to user QoE requires new models of user perception of application performance. Typical QoS metrics, such as Round-Trip Time (RTT) or throughput, are well defined for a single end-to-end path, but today’s Internet services depend on multiple paths and systems. Thus, our models must combine the full picture of a service’s QoS to then map it to users’ QoE. Modeling QoE in itself is challenging, because user perception is subjective and contextual. For example, if RTTs are larger than usual, a user who is doing remote login may feel that the connection is unusable, whereas another who is watching YouTube may notice no problem (because YouTube has a playout buffer to mask some network delay). Second, diagnosing the root cause of poor Internet QoS that degraded QoE requires understanding the complexity of today’s Internet services with little cooperation among different players (e.g., ISPs and content providers), who have no incentives for sharing information to help in narrowing down the cause. Our algorithms must identify the correct explanation with high accuracy based only on data observable from the Internet’s edge.

Résumé du projet de recherche (Langue 2)

The goal of this doctoral thesis is to deliver algorithms, methods, and software systems to measure Internet QoE and diagnose the root cause of QoE impairments. To achieve this goal, we must solve two inference problems. First, the detection problem arises because instrumenting every possible application to collect user engagement or application-level QoS metrics is impractical, so we ask: can we infer QoE simply by monitoring metrics that can be easily collected on the device (e.g., network QoS metrics, CPU activity)? Second, the diagnosis problem comes from the fact that Internet services depend on multiple network paths and services, which are under distributed control. Given that we can only monitor the user’s device and perhaps few other hosts in the Internet, and that monitoring information is noisy, we ask: how to conduct root-cause analysis on partially hidden structures? This thesis will address both of these problems. To detect QoE impairments, we will leverage datasets collected with HostView [1] and YoMoApp [2]. HostView is a measurement tool developed in our team to collect network QoS data annotated with explicit user feedback on network performance, and YoMoApp is an Android application which passively monitors application-layer key performance indicators (KPIs) of YouTube adaptive video streaming on end-user smartphones, along with network traffic statistics and explicit user feedback. The student will investigate how to apply statistical learning models to analyse this data and how to train classifiers that predict whether users are satisfied or not with application performance, simply based on network QoS metrics. While application level monitoring provides very valuable information to predict QoE, it is generally more cumbersome than network-layer monitoring, as not every application provides APIs to access relevant metrics, and in most cases, device root access must be granted to perform measurements deeply into the application, hindering large-scale passive monitoring. QoS metrics at the network layer provide more general information, which is not tied to any particular application, and therefore becomes attractive for large-scale and application-agnostic QoE monitoring. To isolate the root-cause of QoE impairments, the student will first analyze the traces collected from the user’s device to narrow down the set of tests to conduct. The student will experiment with different anomaly detection methods on the different metrics we can extract from packet traces and from the device’s performance and connectivity to identify the hosts affected, the metrics that are behaving anomalously, and the applications affected. The student will then rely on both active probing to pinpoint potential performance bottlenecks and correlation of datasets collected from different vantage points to identify the root-cause. In particular, the student will use distributed measurement platforms such as RIPE Atlas, CAIDA’s Archipelago, and PlanetLab to augment the visibility on Internet-scale issues. References: [1] D. Joumblatt, R. Teixeira, J. Chandrashekar, and N. Tatif. HostView: Annotating end-host performance with user feedback. in Proc. of ACM HotMetrics workshop, 2010. [2] F. Wamser, M. Seufert, P. Casas, R. Irmer, P. Tran-Gia, and R. Schatz. Understanding YouTube QoE in Cellular Networks with YoMoApp – a QoE Monitoring Tool for YouTube Mobile in Proc. of ACM MOBICOM, 2015.
This thesis is in collaboration with Pedro Casas of AIT Vienna.

The candidate should have a strong background in computer networks (TCP/IP and application layer protocols, HTTP(S), in particular) and computer systems performance measurement. The candidate should have previous experience with large-scale network measurements, in particular using distributed monitoring systems as the ones mentioned before. The candidate should have knowledge of data analysis techniques (statistics, data mining, machine learning) and some related tools such as MATLAB, GNU R, scikit-learn, Weka, etc. The candidate should be able to write (scientific writing) and communicate fluently in English.