Paper Summary
Inferring Streaming Video Quality from Encrypted Traffic: Practical Models and Deployment Experience

Sarthak | 2020CS10379

June 2024

Problem Statement

The paper addresses the challenge of inferring video quality metrics (such as startup delay and resolution) for encrypted streaming video services.

Motivation

- ISPs need accurate methods to infer video quality for effective network management and capacity planning
- Increasing use of encryption for video streams (via HTTPS and QUIC) complicates this task
- Previous (deep-packet) approaches unable to extract application level information due to encryption

Key Idea

To use ML regressors like Adaboost, linear, logistic, ridge, SVR, decision tree, and random forest to infer key quality metrics like startup delay and resolution.

Framework

- The primary metrics of interest are startup delay, resolution, bitrate, resolution switching, and rebuffering and features were taken from network, transport, & application layers
- Data was collected using a Chrome extension that monitors application-level information

- A total of 32 models (regressors and classifiers) were trained for each quality metric, using 10-fold cross-validation to ensure accuracy with grid search on hyperparameters
- Existing models (like BUFFEST & Requet) were evaluated and adapted for different video services, this included retraining decision tree-based models

Contributions

- Developed a single composite model that can infer video quality metrics for multiple streaming services (Netflix, YouTube, Amazon, Twitch)
- Improved the granularity of predictions, allowing for precise inference of metrics like startup delay rather than coarse classifications
- Released a dataset of over 13,000 labeled video sessions to the community for benchmarking and further research
- Developed a chrome extension to label traffic traces with the appropriate video quality metrics

Strengths

- Designed to work in real-world deployment settings where video traffic is mixed with other types of traffic, and traffic statistics are collected at coarse granularities due to aggregation
- Models are based on decision trees, so one can retrain them for any service using labeled data, without the need to reverse engineer each individual service

Weaknesses

- Does not generalize well to services not included in the training set
- Errors in identifying video sessions amidst mixed traffic can propagate through the inference process