

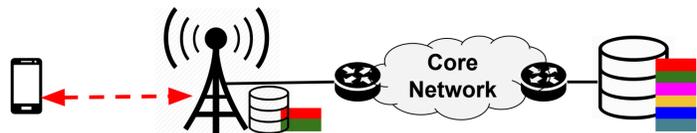
Can Recommenders Compensate for Low QoS?

Mateus Nogueira¹, Daniel Menasché¹, Pavlos Sermpezis², Thrasyvoulos Spyropoulos³

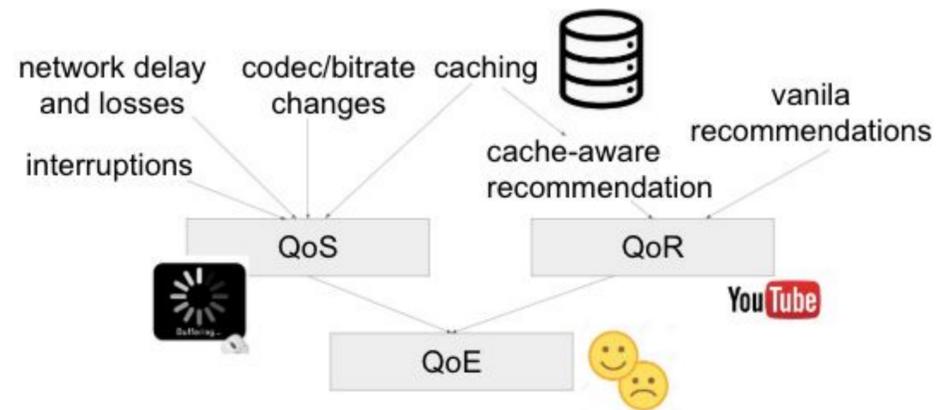
¹ UFRJ, Brazil, ² Aristotle University of Thessaloniki, Greece ³ EURECOM, France

1. Background and Motivation

- High Internet mobile traffic for video services
- Mobile networks struggle to attain high QoE
- A traditional solution: caching
- A recent solution: QoS-friendly recommendations, e.g., bias recommendations towards caches [1, 2, 3]

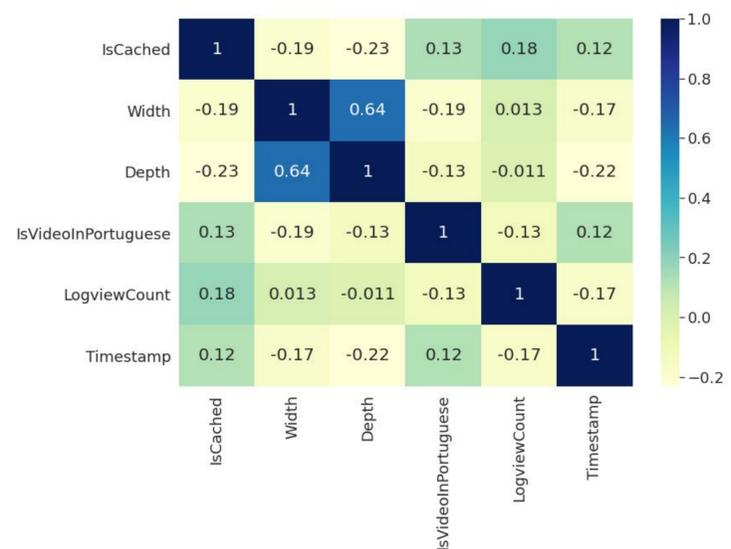


- Promising gains: 10x increase in cache hit ratio [1]
- But, no experiments in real Internet
- **Goal:** assess the role and implications of recommendation systems on QoE with real Internet measurements



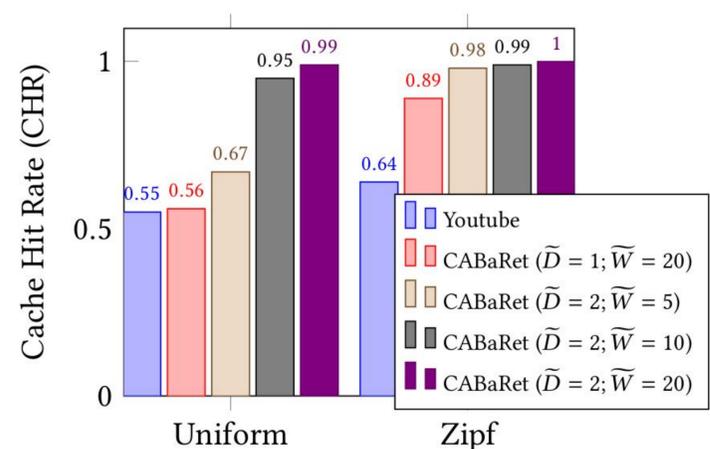
3. Experimental Results

What is the correlation between host distances and content distances? Empirically found significant correlation between host and content distances



Can recommenders compensate for low QoS? Yes!

- 1) Leveraging network measurements, we learned that **recommendation reorderings** are sufficient to increase **cache hit rate (CHR)** from **0.64** to **0.89**
- 2) Allowing for replacements of videos that are at most two hops away in the **recommendation graph** suffices to reach a **CHR** of **0.98**.



- [1] S. Kastanakis et al., "CABaRet: Leveraging Recommendation Systems for Mobile Edge Caching", in ACM MECOMM (SIGCOMM workshop) 2018.
 [2] P. Sermpezis et al., "Soft Cache Hits: Improving Performance through Recommendation and Delivery of Related Content", in IEEE JSAC, 2018.
 [3] Kastanakis, S. et al., "Network-aware recommendations in the wild", in IEEE Transactions on Mobile Computing, 2020..

Contact us at: mateussnogs@gmail.com, sadoc@dcc.ufrj.br, sermpezis@ics.forth.gr, spyropou@eurecom.fr

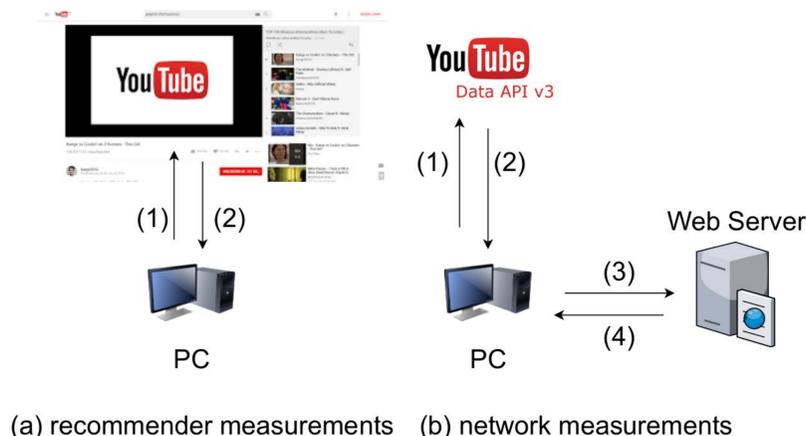
2. Experimental Setup

Goal: quantify interplay between (i) QoS and (ii) content recommenders

→ ultimately, build better recommendation systems

Two key concepts: content and host distances

	Content distance	Host distance (near cache?)
Infer properties of:	Youtube recommender graph	Delay and losses towards contents
Measurements leverage:	Youtube API	Ping and traceroute
Unit:	Hops in graph	Milliseconds



Challenges and solutions:

a) Youtube API quota on number of requests/day.

Solution: breadth first search through recommender graph starting from trending topics

b) Delay to access contents depends on vantage point.

Solution: measure delays using ping and traceroute towards Youtube servers from multiple vantage points.