JOINT OPTIMIZATION OF CACHING & RECOMMENDATIONS

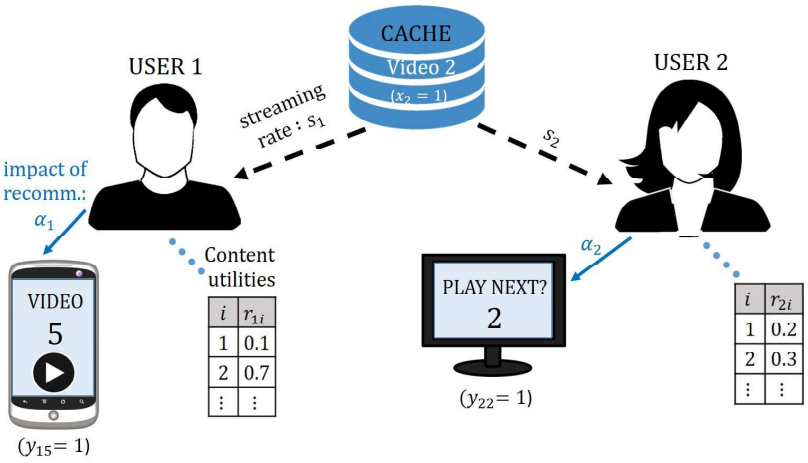
DIMITRA TSIGKARI AND THRASYVOULOS SPYROPOULOS
firstname.lastname@eurecom.fr

BACKGROUND & MOTIVATION

- Google & Netflix can control both caching and recommendation systems.
- Quality of Experience (QoE) depends on both Quality of Service (QoS) and Quality of Recommendations (QoR) [2].

Given the network topology and the content utilities, what to cache (variable X) and what to recommend to every user (variable Y)?

- Algorithms proposed in the literature [3, 4] do not solve the problem jointly!



PROBLEM

$$\underset{X,Y}{\text{maximize}} \sum_{\text{users}} \overbrace{\underbrace{\text{QoS}}_{\text{expected rate}} + \beta \underbrace{\text{QoR}}_{\text{recomm. content utilities}}}^{\text{QoE}}$$

subject to: cache capacity and number of recommended items (N).

Explanations:

- $\text{QoS} = \sum_i \frac{\alpha_u}{N} y_{ui} s_u(X, i) + (1 - \alpha_u) p_{ui} s_u(X, i)$.
- $\text{QoR} = \sum_i y_{ui} \log(r_{ui})$.
- $\beta > 0$ captures the importance of each factor.

→ **This problem is NP-hard!**

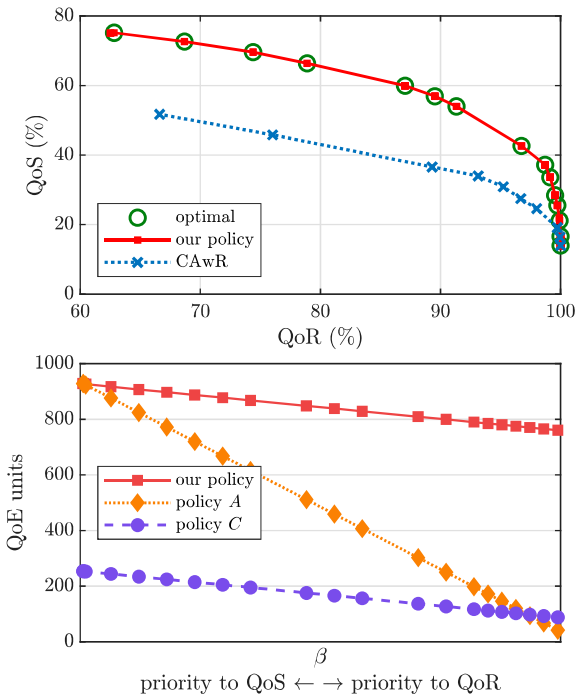
METHOD

$$\underset{X,Y}{\text{maximize}} \sum_{\text{users}} \text{QoE} = \underbrace{\max_X \Big(\underbrace{\max_Y \sum_{\text{users}} \text{QoE}}_{\text{inner problem}} \Big)}_{\text{outer problem: submodular!}}$$

- For any cache X, the inner problem can be solved *optimally*.
- The resulting function (of variable X) is *submodular* and monotone increasing!
- Our (greedy) algorithm achieves a constant approximation guarantee: $1/2$ for equal-sized contents.
- Our algorithm: It starts with empty cache and in every round of selection, it caches the content that maximizes the marginal gain while it solves the inner problem (which gives the recommendations Y).

PERFORMANCE

- The theoretical approximation guarantees are validated.
- The trade-off (Pareto) curve of our algorithm dominates other state-of-the-art algorithms (CAwR in [4]).
- Our algorithm outperforms baseline non-joint policies in terms of achieved QoE.



Baseline policies A and C cache the most popular contents. Policy A recommends only cached contents while policy C recommends the contents with the highest utility per user.

REFERENCES

[1] D. Tsigkari and T. Spyropoulos. User-centric optimization of caching and recommendations in edge cache networks Available at http://www.eurecom.fr/~spyropou/papers/QoE-joint-extended.pdf
[2] P. Sermpezis et al. Towards QoS-Aware Recommendations. In Arxiv (2019)
[3] T. Giannakas et al. Show me the Cache: Optimizing Cache-Friendly Recommendations for Sequential Content Access. In WoWMoM 2018
[4] L.E. Chatzieleftheriou et al. Jointly Optimizing Content Caching and Recommendations in Small Cell Networks. In IEEE Trans. Mob. Comput. (2019)

INTERESTED?

You can check out our article at