Are Netflix Videos Edge-Cacheable? Exploration of a Deep Learning based Prefetching Strategy using a Real-World Dataset

Shruti Lall and Raghupathy Sivakumar
{slall,siva}@ece.gatech.edu
Georgia Institute of Technology

1 Introduction

Internet traffic load is not uniformly distributed through the day - it is significantly higher during peak-periods, and comparatively idle during off-peak periods. An interesting question we consider in this work is: *could the peak-period load be shifted to off-peak periods using intelligent prefetching?* We specifically focus on Netflix since it now contributes to the largest percentage of global Internet traffic by a single application [1]. Using a real-world dataset of Netflix viewing activity that we collected from 523 users (using Amazon mTurk [2]) spanning a 1-year period and comprised of $1,021,917$ titles, we explore if it is possible to accurately anticipate what Netflix content a user is likely to consume in the future, and cache it at the edge closest to the client (at an end-device or a server attached to an access point e.g. a WiFi router) during off-peak periods. We restrict the scope to Netflix series and documentaries that together account for 70% of a typical user's Netflix load in terms of bytes fetched. To this end, we design a deep learning based prediction algorithm that prefetches Netflix content ahead of time based on the user's past viewing patterns. We show that the algorithm is able to achieve a prediction accuracy of nearly 80%, thus indicating that Netflix content is indeed predictable for edge-caching.

2 Deep Learning Prefetching Strategy

We design and implement a deep learning based prefetching (DLP) algorithm which, given the watch history of a user for the past 24 hours, predicts how many of the next episodes of the series to prefetch for future watch-sessions. We prefetch the episodes during the off-peak hours; the off-peak period is defined to be 2am to 6am, and the peak period is defined to be 6am to 12am, and 12am to 2am [3]- [5]. The DLP algorithm models the prediction problem as a sequence problem, where the user's past viewing behavior is encapsulated as a sequence of the number of episodes they watch from series in their past watch-sessions. An LSTM network is chosen as the learner for the predictions. Furthermore, for each series that episodes are being prefetched for, separate LSTM networks pertaining to features for that particular series are also used for the prediction. The following features are considered: genre, rating, runtime, year, total number of season in series, day of week for which the content is being prefetched for, and number of series watched in the same watch-session. The outputs from the various LSTM networks are combined using an adaptive boosting algorithm. A prediction of how many episodes the user will watch in proceeding watch-sessions is then made.

3 Preliminary Results

We evaluate the DLP algorithm on the data collected for 3 different cache eviction policies in terms of the prediction accuracy (PA) and cache efficiency (CE). Once the algorithm predicts the number of episodes to prefetch, the prefetched episodes are stored in the cache with an associated time-to-live (TTL) parameter based on the cache eviction policy. The policies that we consider are:

- Simple eviction: The cache is emptied at the end of each watch-session before any new content is prefetched.
- Per-user average: For each user, the average number of watch sessions between when the content is prefetched, and when it is actually watched is computed; this value is used as the TTL for each item that is prefetched.
- Maximum watch sessions (WSs): The TTL parameter is set to 20 for each item in the cache that is prefetched.

The PA and CE is shown in Fig. 1. We are able to achieve the highest PA of 79.6% with the max WSs eviction policy, however, this is at the cost of the lowest CE of 51.4%. The simple eviction policy performs the worst in terms of accuracy with a PA of 64.2%; this is because approximately 24% of prefetchable content (computed over our entire dataset) is not consumed in the immediately proceeding watch-session.

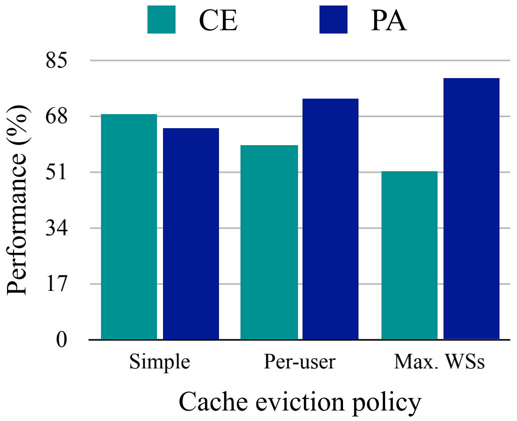

Figure 1: Prefetching algorithm results across all users

The per-user average eviction policy performs better with a PA of 73.4% and a CE of 59.1%. The cache eviction policy can be appropriately tuned depending on whether a strong preference for a high PA is desired, or if a conservative CE is desired.

Acknowledgements

This work was supported in part by the Wayne J. Holman Endowed Chair and the National Science Foundation under grants CNS-1813242 and CPS-1837369.

References

[1] (2019) 2019 global internet phenomena. [Online]. Available: https://www.sandvine.com/press-releases/sandvine-releases-2019-global-internet-phenomena-report

[2] (2019) Amazon mechanical turk. [Online]. Available: https://www.mturk.com

[3] A. Ghosh, R. Jana, V. Ramaswami, J. Rowland, and N. K. Shankaranarayanan, "Modeling and characterization of large-scale wi-fi traffic in public hot-spots," in *2011 Proceedings IEEE INFOCOM*, April 2011, pp. 2921–2929.

[4] K. Fukuda, K. Cho, and H. Esaki, "The impact of residential broadband traffic on japanese isp backbones," *SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 1, pp. 15–22, Jan. 2005. [Online]. Available: http://doi.acm.org/10.1145/1052812.1052820

[5] S. Sundaresan, W. de Donato, N. Feamster, R. Teixeira, S. Crawford, and A. Pescapè, "Broadband internet performance: A view from the gateway," in *Proceedings of the ACM SIGCOMM 2011 Conference*, ser. SIGCOMM '11. New York, NY, USA: ACM, 2011, pp. 134–145. [Online]. Available: http://doi.acm.org/10.1145/2018436.2018452