# Poster: *While You Were Sleeping*: Time-Shifted Prefetching of YouTube Videos to Reduce Peak-time Cellular Data Usage

Shruti Lall, Uma Parthavi Moravapalle and Raghupathy Sivakumar
Georgia Institute of Technology, Atlanta, GA, USA
{slall,parthavi,siva}@ece.gatech.edu

## ABSTRACT

The load on wireless cellular networks is not uniformly distributed through the day, and is significantly higher during peak-times. In this context, we present a time-shifted prefetching solution that prefetches content during off-peak periods of network connectivity. We specifically focus on YouTube as it represents a significant portion of overall cellular data-usage. We make the following contributions: first, we establish that a significant portion of a user's YouTube watch behavior is indeed predictable by analyzing a real-life dataset of YouTube watch history spanning a 1-year period, from 206 users comprised of over 1.8 million videos; second, we present a prediction algorithm called *MANTIS* using a K-nearest neighbor classifier approach and show that the algorithm can reduce the traffic during peak-times by 34% for a typical user.

## CCS CONCEPTS

• **Networks** → Mobile networks; • **Computing methodologies** → Classification and regression trees.

## KEYWORDS

Edge caching; KNN classification; Prefetching

**ACM Reference Format:**
Shruti Lall, Uma Parthavi Moravapalle and Raghupathy Sivakumar. 2019. Poster: *While You Were Sleeping*: Time-Shifted Prefetching of YouTube Videos to Reduce Peak-time Cellular Data Usage. In *Los Cabos '19: ACM International Conference on Mobile Computing and Networking, Oct 21-25, 2019, Los Cabos, Mexico.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

## 1 INTRODUCTION

Wireless spectrum is expensive. The federal communications commission's (FCC) AWS-3 auction (in 1700MHz and 2100MHz blocks) netted approximately \$45B for 65MHz of spectrum (at \$2.71 per MHz-POP[1]), with AT&T being the highest bidder at \$18.2B followed by Verizon at \$10.2B. Wireless service providers upgrade their infrastructure and add spectrum in reaction to load characteristics on their networks. It is typical for upgrades to be triggered when there is a sustained peak usage that exceeds 80% of capacity [1].

To address the peak load conditions and hence defer consequent upgrades, we consider the strategy of *time-shifted prefetching* to address the peak load conditions. We restrict the focus of this work to a specific application - *YouTube*, and explore the time-shifted prefetching of videos to the mobile device so that the videos do not have to be fetched when watched during peak periods. YouTube videos reportedly account for 38% of a mobile user's cellular data usage [2]. This represents the largest share of the cellular bandwidth usage among all applications on the mobile device. Therefore, strategies to prefetch YouTube videos during off-peak periods can have a meaningful impact on the overall peak usage of the cellular network. Thus, the key question we answer is the following: *For a given wireless user, can YouTube videos be prefetched during off-peak periods so that the data usage for fetching videos during the peak periods is reduced?*

## 2 CELLULAR DATA USAGE VARIATIONS ACROSS TIME-OF-DAY

Traffic load on mobile networks is not uniformly distributed through the day and is significantly higher during peak periods. To exemplify this, we performed a bandwidth probe with a Google Pixel smartphone (with Android Pie), and measured the available bandwidth over a T-Mobile cellular network. The probe was done by running a speedtest that downloads a small file from a web server to the mobile device, and using the download time to estimate the throughput. The speedtest was conducted every 30 minutes on the Android device for 5 consecutive days, while the device was connected to a cellular network; Fig. 1 shows the average of the measurements across 5 days. We notice that the available bandwidth varies with time of day. Specifically, we observe an increase in the available bandwidth between 2 AM to 5 AM , and a subsequent decrease from 6 AM to 8:30 PM and a gradual increase from 8:30 PM. This indicates that the traffic load varies through the course of the day i.e. low data rates correspond to high traffic, and vice-versa. Similar trends have also been shown in other cellular traffic distribution studies [3]. There is, thus, a potential to utilize the available bandwidth during off-peak periods, for prefetching video content.

[1] MHz passing one person.

*Los Cabos '19, Oct 21-25, 2019, Los Cabos, Mexico*

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…\$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

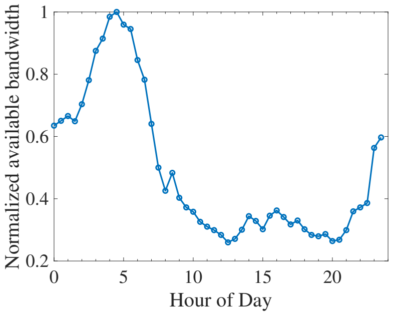

**Figure 1: Available bandwidth across the day**

# 3 ON THE PREDICTABILITY OF YOUTUBE WATCH BEHAVIOR

## 3.1 Data Collection

Successful prefetching of YouTube videos requires the ability to predict what videos the user is likely to watch in the future. In order to study the feasibility of prefetching, we perform an analysis on a dataset comprised of YouTube usage history. To collect the dataset, we rely on Amazon Mechanical Turk (mTurk) to gather anonymized watch history from users [4]. The mTurk task posted required users to navigate to Google's *Takeout* page and access their YouTube related data, which can be retrieved as an archive file that contains files related to their watch-history, playlists and subscriptions data.[2] In the collected dataset, there are $1,798,132$ videos watched by 206 users, and the total number of *unique* videos watched by the users is $1,116,271$ videos. To understand if the mTurk users were representative of typical YouTube users, we also collected demographics information for the participants in the study including country, gender, age, and education level. An analysis of the demographics shows that the participants are indeed representative of YouTube's published statistics on its users [5]. To further overcome any biases in the mTurk dataset, we also show performance results later for an independently collected dataset used by Park *et al.* in [6].

## 3.2 Predictability of Watch Behavior

*3.2.1 Watch-history:* A simple approach for prefetching is to consider the past watch-history of the user and to select videos from the watch-history to prefetch. Such an approach would work if there is repetition in the user's watch behavior. For our collected dataset, we find that, on average, only 13% of videos are re-watched by the user; thus, the potential for effective prefetching of videos simply based on what a user has previously watched is low.

*3.2.2 Related Videos:* YouTube algorithmically determines videos that are related to one another using the video's metadata, and also collaborative filtering methods. The related videos are also utilized by YouTube's powerful recommendation engine. Here we explore the possibility of utilizing *related videos* as the set of videos from which to prefetch. We use YouTube API's *relatedToVideoId* endpoint to retrieve a list of videos which is related to a particular video. For a particular user, we fetch 50 related videos of every video that has been watched by the user, and then see if any of the related videos were watched later. **We perform this analysis for all the users in our collected data set and found that 59% (standard deviation of 16%) of all videos watched by the user were in the related videos set of previously watched videos.**

# 4 *MANTIS*: PREFETCHING ALGORITHM FOR YOUTUBE VIDEOS

Based on the insight that a user's watch behavior is predictable based on the related videos set of previously watched videos, we propose a prefetching algorithm *MANTIS* that accurately predicts the videos a user will watch from her related videos set, while ensuring an acceptable prefetching efficiency.

## 4.1 Data Preprocessing and Feature Design:

The intelligence in *MANTIS* stems from a machine learning classifier that predicts the videos a user is likely to watch given the patterns in their viewing behaviour from the past. To train the classifier, *MANTIS* uses 15 features associated with every video in the set of videos from which to prefetch (*fetch set*), these are: retrieval date (date of when the video from which the related videos are obtained, was watched), time difference (difference between video upload day and prefetching day), number of views, number of likes, number of dislikes, number of comments, channel ID (channel which uploaded the video), category ID (the video category ID), number of channel subscribers, number of channel uploads, subscribed (whether the user has subscribed to the channel), repeats (the number of times the video has been seen by the user in the past), video duration, playlist (whether the video appears in any user playlists), and video tags. Given that there is a sizeable amount of data to be considered for accurate prediction of videos a user may watch in the future, utilizing all the features is computationally expensive. *MANTIS* eliminates this redundancy with the application of principal component analysis [7].

## 4.2 Intelligent Prefetching Algorithm:

Using the aforementioned features, *MANTIS* uses a K-nearest neighbours (KNN) classification algorithm to classify the vidoes to prefetch from the fetch set. This classifier is trained over 90 days (*training-period*), and the related videos are fetched from videos watched by the user over the past 14 days (*fetch-period*). The *training-period* was empirically determined by evaluating the performance of the KNN classifier for a 30 day test period for each user's entire watch-history, while the *fetch-period* was found to be optimal at 14 days. Once we obtain the 50 related videos for every video watched during the *fetch-period*, we further reduce the size of the fetch

[2] We were advised by the IRB that IRB approval was not required as no private or personally identifiable information was collected.

set by selecting videos in the users most preferred 30 channels and 3 categories. After this, each video in the fetch set is classified as belonging to one of two classes, namely, *chosen* or *discarded*. *Chosen* means that the video is likely to be watched in the near future by the user, and should be prefetched. *Discarded* means that the video is not likely to be watched by the user, and therefore, should not be prefetched.

## 5 PRELIMINARY PERFORMANCE RESULTS

We define the following metrics for evaluating the *MANTIS* classification algorithm: *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), *False Negative* (FN), *Prefetch Accuracy (PA = TP/TP+FN)*, *Prefetch Efficiency (PE= TP/TP+FP)*, and *Overall Accuracy (OA=(TP+TN)/(TP+FP+TN+FN))*. To evaluate *MANTIS*, the data is first parsed and stored in a Postgres PSQL database. The KNN classifier is implemented using Python on a macOS system with a 2.5 GHz Intel Core i7. The parameter values used for evaluating *MANTIS* are: *K*= 3 (for KNN), number of users= 206, *fetch-period*= 14 days, *training-period*= 90 days, and prefetching time=4 am. *MANTIS* was evaluated for 30 continuous viewing days where for every user, for each day, *MANTIS* predicts what the user is likely to watch during the day, and fetches it at 4am.

The impact of *MANTIS* can be shown in terms of the bandwidth (BW) consumption for the users. When *MANTIS* is implemented, and videos are prefetched during off-peak hours, there is a decrease in the BW consumed by the users in the rest of day. This decrease corresponds to a smoothening of the network traffic demand curve. The BW reduction per user during peak periods, is shown in Fig. 2; this is shown as a function of the number of videos watched by the user over their test period. On average, a BW savings of 3.3 GB across the 206 users is observed. Fig. 3 summarizes the per-user BW consumption for peak and off-peak periods, with and without the use of *MANTIS*, across all users. We see that *MANTIS* is able to achieve a peak-time BW reduction of 34% while increasing the overall BW consumption, during off-peak periods, by 12% (from 10.6 Gb to 11.9 Gb).

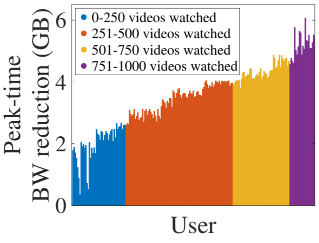

**Figure 2: Peak-time BW reduction across 206 users for 1 month**

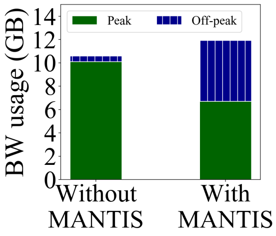

**Figure 3: Average BW usage for 206 users for 1 month**

In addition, using the above-defined parameters, the results of the prefetching algorithm, averaged over the 30 day testing period, for all 206 users is shown in Fig. 4.

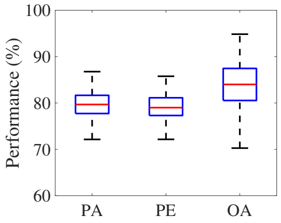

**Figure 4: Performance of *MANTIS* across 206 users**

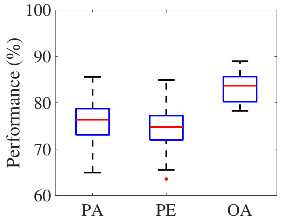

**Figure 5: Performance of *MANTIS* on dataset in [6]**

We also evaluate *MANTIS* on a dataset that was collected by Park *et al.* [6]. The duration over which the collected user's history was monitored varies from 2 weeks to 14 weeks; we applied *MANTIS* to 57 users whose watch-history spanned at least 5 weeks. We trained *MANTIS* for 30 days for these user, and predicted videos for 1 week thereafter. We found that *MANTIS* performs well for this dataset with PA= 76.2%, PE= 74.3%, and an OA= 83.1%, as shown in Fig. 5.

## 6 CONCLUSIONS

To conclude, we address the problem of the high variance between off-peak and peak network traffic, through a time-shifted prefetching strategy for YouTube content. A dataset containing the watch-history of 206 users was used to study YouTube watch behavior and used to aid in the the development of the prefetching strategy that relies on prefetching videos related to content that has been previously consumed by the user. A KNN classifier, after being empirically tuned, was used for evaluating the prefetching algorithm across the users, and also compared to data collected by different authors. We found that an overall reduction of 34% in traffic during peak periods was achieved through this algorithm, while increasing the overall traffic consumption by 12%.

## ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under grants CNS-1513884, IIP-1701115, CNS-1813242, and CNS-1837369, and the Wayne J. Holman Chair.

## REFERENCES

[1] (2019) 4 ways service providers can improve capacity forecasts. [Online]. Available: https://www.sevone.com/white-paper/4-ways-service-providers-can-improve-capacity-forecasts

[2] (2019) Sandvine 2019 mobile internet phenomena. [Online]. Available: https://www.sandvine.com/2019-mobile-internet-phenomena-report

[3] (2016) Sandvine 2016 global internet phenomena. [Online]. Available: https://www.sandvine.com/blog/2016/06/global-internet-phenomena-report-2016-latin-america-north-america

[4] (2019) Amazon mechanical turk. [Online]. Available: https://www.mturk.com

[5] (2019) The latest youtube stats on audience demographics: Who's tuning in. [Online]. Available: https://www.thinkwithgoogle.com/data-collections/youtube-viewer-behavior-online-video-audience/

[6] M. Park, M. Naaman, and J. Berger, "A data-driven study of view duration on youtube," in *ICWSM*, 2016.

[7] H. Abdi and L. J. Williams, "Principal component analysis," *WIREs Comput. Stat.*, vol. 2, no. 4, pp. 433–459, Jul. 2010. [Online]. Available: https://doi.org/10.1002/wics.101