User profiling for Pre-fetching or Caching in a Catch-Up TV Network

Andersson, Jens A; Höst, Stefan; Kihl, Maria; Arvidsson, Åke; Du, Manxing; Zhang, Huimin; Lagerstedt, Christina

Published in:
IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, BMSB

DOI:
10.1109/BMSB.2016.7521984

2016

User profiling for Pre-fetching or Caching in a Catch-Up TV Network

Jens A Andersson*, Åke Arvidsson†‡, Manxing Du§, Huimin Zhang§, Maria Kihl*, Stefan Höst*, Christina Lagerstedt§
*Department of Electrical and Information Technology, Lund University, Sweden
†Ericsson AB, Sweden
‡Department of Computer Science, Kristianstad University, Sweden
§Acreo Swedish ICT AB, Sweden

{Jens_A.Andersson, Maria.Kihl, Stefan.Host}@eit.lth.se,
Ake.Arvidsson@ericsson.com, Ake.Arvidsson@hkr.se,
{Christina.Lagerstedt}@acreo.se

***Abstract*—We investigate the potential of different pre-fetching and/or caching strategies for different user behaviour with respect to surfing or browsing in a catch-up-TV network. To this end we identify accounts and channels associated with strong or weak surfing or browsing respectively and study the distributions of hold times for the different types of behaviour. Finally we present results from a request prediction model and a caching simulation for the different types of behaviour and find that the results are relatively similar.**

I. Introduction

A provider of a service or product should strive to keep the customer satisfied. In the context of IPTV and Video on Demand (VoD) this includes providing as low a latency as possible between user action and service delivery. Reducing the *zapping time*, i.e. the time from channel selection to start of playout, is one example, quick reaction to fast forward or rewind another. A solution is to predict the user's next action and prefetch larger or smaller program parts accordingly. The challenge is to predict the correct content, but also to prefetch the right amount of this content so that the zapping time is as low as possible while the continued playout will be performed without glitches. To download data that is never used is of course a waste of resources.

When browsing or zapping, quickly switching from one TV channel to another, a user initiates many program views but consumes only a small part. Reducing the zapping time in IPTV services is essential for the user's perceived quality (QoE) [7]. Reducing unnecessary downloads of unseen parts of programs is also of importance not only for fixed access networks but also in mobile networks [6][4].

VoD services have been studied regarding both prefetching and caching, but using the same strategy for all subscribers. Users groups' variant behaviour, specially regarding zapping, is expected to influence the network's prefetching and caching strategies. Very little has previously been studied regarding prefetching and caching strategies adaption to user profiles in VoD networks. This study tries to narrow this research gap by studying users as well as channels in this context. Two user groups, *zappers* and *loyals* are identified and their impact on prefetching and caching in a so called Catch-Up TV network is analysed. Furthermore, differences between zapping prone and loyal prone channels are presented. Zappers changes quickly from program to program searching for the wanted view while loyals consumes the first selection to the end before requesting another view.

978-1-4673-9044-6/16/$31.00

The studied operator's service is to offer channels' subscribers with the option of watching programs in retrospect. Users in the studied network subscribe to some, but not necessarily all, of the available channels. Their selection of views is for series and episodes rather than channels.

Earlier studies have analysed zapping in broadcast or IPTV networks. Cha *et al.* have analysed user behaviour in an IPTV network [2]. Three user modes were defined and analysed, *surfing* which is the equivalent to what herein is called zapping, *viewing*, and *away*. In [5] Gopalakrishnan *et al.* are modelling the behaviour of a single user - the Couch Potato - viewing session with events like Start, Pause, Play, Fast Forward, and Rewind. From state Play 63% of all state changes go into Fast Forward, an indiciation of zapping behaviour. Ali-Edin *et al.* discusses the *impatient user* behaviour in [1]. In their study over 90% of all sessions last less than one hour, and 20% less than 30 seconds. A comparison with Yu *et al.* [8] shows the same behaviour, except for views lasting longer than 10 minutes. The impact on prefetching and caching is not studied.

Zapping in a broadcast or IPTV network means changing of channels. In a catch-up TV network or a play service zapping have another modus, users tend to follow series and switches between episodes rather than channels. The analysis by Du *et al.* in [3] is focused on pre-fetching in a Video on Demand service but has also looked into the viewing time per request. 20% of all requests are shorter than two minutes, which can be defined as zapping. The impact from zapping on caching and/or pre-fetching strategies has also been studied by Du *et al.* [3] and Zhang [9]. In these studies no categorisation of users were done.

II. Identifying and Discussing Zappers and Loyals

The dataset from the IPTV operator consists of logged and time stamped request events. It contains over 17 million requests from about 570,000 accounts viewing more than 80 TV channels. Only logged data from what can be denoted as *active accounts* were included. An active account is defined by belonging to the 30% of all individual accounts that contribute to 75% of all requests, see Figure 2.

The analysed dataset lacks logging of when a user stops viewing a program or even starts viewing it; Only the time when the user requested a specific program is recorded. The *program hold time* needed for the zapping ratio thus had to be estimated.

$$p_i = \min\{(r_{i+1} - r_i), (e_i - s_i)\}, i = \{1, \ldots, (l-1)\}$$

p_i is the program hold time for request i within one session, r_i is the time request i is made and l is the number of requests in the viewing session. s_i and e_i is the start and end hour of when the requested program i was originally broadcast.

p_i for $i = l$, that is the last request in a session has to be an educated guess, if not omitted. Omitting the last request in each session has effect on the number of requests per session, and the sessions will be shorter and more frequent since the majority of the last requests has to have an estimated duration of a full program view. In our case the study is aimed at browsing and thus this choice has little or no effect.

In Figure 1, the frequency of program hold times during the measurement period and the viewed program's full duration are plotted in a log-log diagram. The graph resembles one found in [2] where the frequency of channel hold time is plotted. The main difference is that in [2] TV channels are studied. TV channels have more or less infinite duration. Programs on the other hand are finite, which of course has an impact on the program hold time; the latter cannot be longer than the former. The two minor peaks between 10 minutes and 1 hour and the greater gradient of the second part of the plot are explained by this fact.

From the peak at 40 seconds and up to approximately 15 minutes the graph in Figure 1 follows a power-law distribution. The referenced plot in [2] has the peak at channel hold time 4 seconds, while in Figure 1 the peak is found at program hold time 40 seconds. One minute, as suggested by Cha *et al* and others, as an upper delimiter for zapping cannot be used here. Instead 300 seconds is proposed as more appropriate for this study.

A *zapping ratio* was calculated according to

$$R_a = \frac{Z_a}{N_a}$$

where Z_a is number of requests shorter than 300 seconds, herein called zapping request, and N_a is the total number of requests for an active account a.

Figure 3 shows the zapping ratio CDF for all users in the active users list. These users were then divided into the two user groups, zappers and loyals, according to the zapping ratio. The top 10% accounts with highest zapping ratio were designated as zappers (17385 accounts) and the 10% accounts with lowest zapping ratio as loyals (19899 accounts). The expressed gradients in Figure 3 point out that per each full program view the user makes one, two or three "zapping" requests.

Users are normally loyal to a few channels, see Figure 4, in

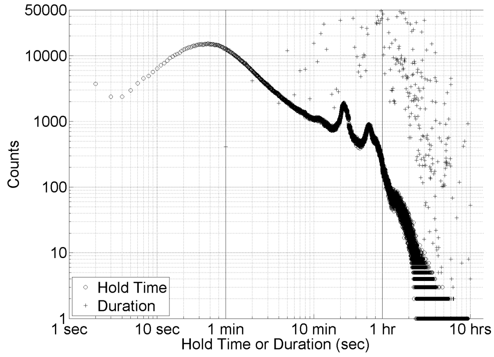

Figure 1. Occurrences of program hold times and requested program's duration

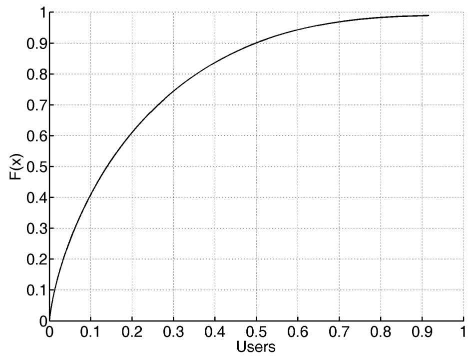

Figure 2. CDF of user's contribution to the total of number of request. The users are sorted descending regarding number of request over the full period.

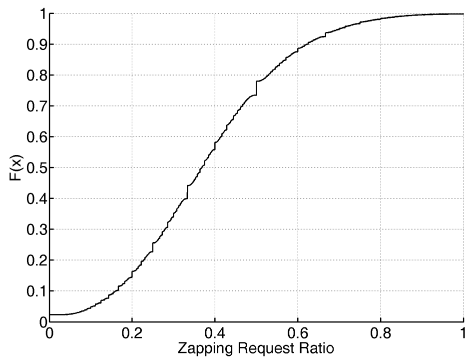

Figure 3. CDF of zapping ratio for all active users. The last request, the one that might have shown the program the user actually wanted to see, is not included.

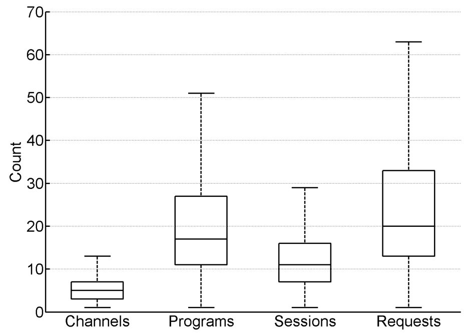

Figure 4. Number of channels, programs, sessions and requests per user. Outliers are removed.

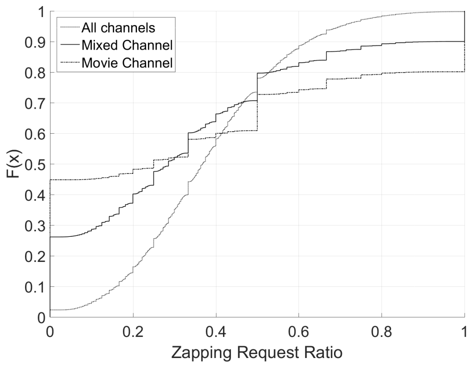

Figure 5. CDF of zapping ratio for two different channels.

this case limited by the sets of subscribed channels. From Figure 5 we can also see that the there is a difference in the zapping ratio distribution between two of the channels in the network, one movie channel and one with mixed content. All requests for the specific channel is evaluated per account. For the movie channel 45% of all accounts have no zapping request, while for the mixed channel the same behaviour is found in approximately 25% accounts. It is therefore of interest to compare also channel's request patterns. The channels were sorted according to zapping prone account frequency. The top 10% channels with the highest number of zapping prone accounts (9 channels) and the 10% of the channels with lowest number of zapping prone accounts (7 channels) where grouped.

As can be seen in Table I and Table II zappers are more prone to watch several different streams per session than loyals. Zappers are more impatient and have shorter sessions. The requests per session average is nearly the same for the two user groups, but the standard deviation for zappers is double that for loyals. The difference in session length is pronounced when comparing zapping prone and loyal prone channels.

A note on self-fulfilling: zappers have been defined to have a high ratio of requests with program hold times ≤ 300 seconds. It is obvious that this definition by intention introduces a bias regarding the number of shorter program hold times for the two user groups, see Figure 6 at 300 seconds. It should also be

Table I
AVERAGE AND STANDARD DEVIATIONS OVER ALL SESSIONS FOR ZAPPERS AND LOYALS.

	Zapping Accounts	**Loyal Accounts**
Session duration (s)	2434.99 ± 1732.46	3171.64 ± 2463.81
Requests/session	1.92 ± 2.09	1.44 ± 0.92
Episodes/session	1.69 ± 1.48	1.37 ± 0.82
Episodes/Requests	0.95 ± 0.14	0.97 ± 0.11

Table II
AVERAGE AND STANDARD DEVIATIONS OVER ALL SESSIONS FOR ZAPPING AND LOYAL PRONE CHANNELS.

	"Zapping" Channels	**"Loyal" Channels**
Session duration (s)	1935.19 ± 1216.61	5703.35 ± 3554.37
Requests/session	1.79 ± 1.62	1.37 ± 1.00
Episodes/session	1.61 ± 1.19	1.23 ± 0.64
Episodes/Requests	0.95 ± 0.13	0.95 ± 0.16

noted that removing "noise" by excluding non active accounts (see section II) might impact on the number of loyals.

III. PRE-FETCHING AND CACHING

In [9] Zhang analyses the next events for one specific channel. By repeating these measurements for zappers and loyals separately over all channels in the network it is possible to make an analysis regarding different gains of prediction of episodes of the same series.

Figure 7 reflects the probability of a user watching episode X of a series as next event requests episode $X + i$ where $i \in \{-10, +10\}$. $i = 0$ corresponds to the cases where the user request the same episode again. Note that it is not possible to see from the recorded data where in an episode the user starts viewing. Events corresponding to $X + 0$ could very well be cases where the user has paused the viewing for a longer period.

Loyals have a higher tendency than zappers to go from episode X to episode $X + 1$ as shown by Figure 7. Zappers on the other hand requests previous episodes more likely than loyals. This could indicate that loyals are more pro-active by

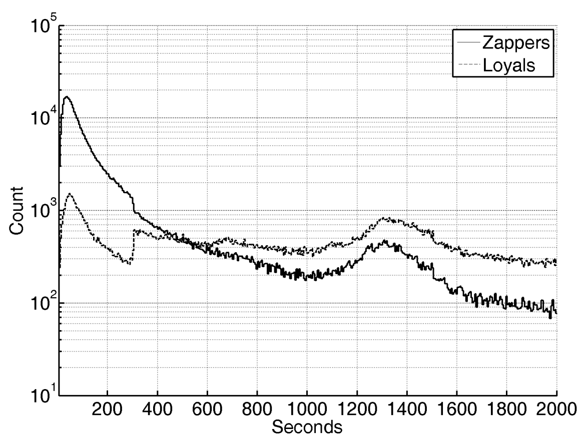

Figure 6. PDF for Program Hold Times. Note the logarithmic scale on the Y axis.

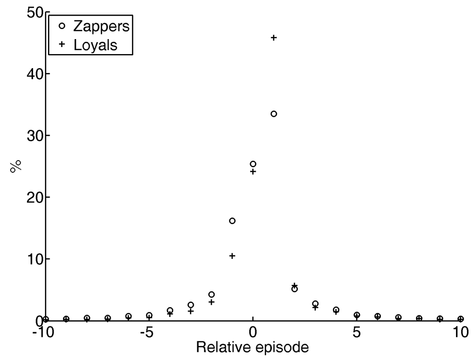

Figure 7. Next view event for zappers and loyals.

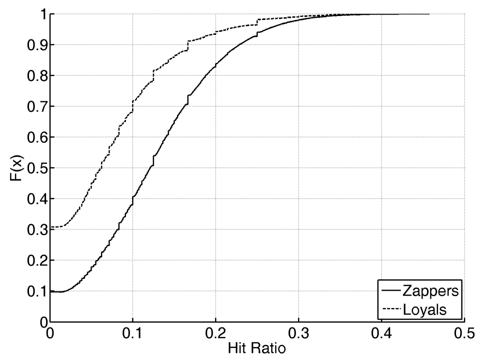

Figure 8. Personal cache hit ratio for zappers and loyals.

making more founded viewing choices, while zappers could to a higher extent be said to be more re-active and just pick a reasonable episode and change if the choice was wrong.

A notable remark is that in the current study more than 53% of the next viewing event for zappers is for another series than the current view and 45% for loyals; Zappers are less loyal to series.

Zappers gain slightly more from personal caching than loyals, Figure 8. The mean hit rate for zappers is 0.25 ± 0.16 and 0.15 ± 0.15 for loyals. This is in accordance with the findings in Table I; Zappers have more request per episodes than loyals.

IV. Conclusion

According to [2] the median and average *channel* hold times are very different: 8 seconds and 14.8 minutes respectively. In this study the median *program* hold time is 11.0 minutes and the average is 20.3 minutes. The zappers in this study can thus be said to be more patient, or perhaps more informed before starting the session, compared to IPTV subscribers.

Overall, 80-85% of the sessions contain only one or two program requests, this indicating that most of the users are informed of what content is requested before starting a viewing session. Even so the analysis indicates that identifying the two user groups zappers and loyals is beneficial. Zappers, as being more impatient than loyals, can gain from a shorter play-out delay. Zappers can also benefit from prefetching of not only episode $X + 1$ but also episode $X - 1$. Zappers also show more value in personal caching than loyals.

Zappers are less loyal to series, but if this implies that zappers also are less loyal to channels is a matter for future studies. Note, that users subscribe to a limited selection of channels, which limits their possibility to show channel disloyalty.

Not only can zapping prone subscribers be identified but also zapping prone channels exist. Thus, profiling of both subscribers *and* channels enabling adaption of e.g. playout delay and prefetching and caching strategies per profile is suggested as a mean of increasing subscriber QoE and satisfaction. The outcome of introducing profiling has to be analysed further in future work.

This work has been funded by Sweden's Innovation Agency VINNOVA and Celtic-Plus/Eureka in the project Next Generation Over-the-top Multimedia Services (NOTTS), and by EIT Digital in the project Networks for Future Media Distribution (NFMD). Maria Kihl is a member of the Lund Center for Control of Complex Engineering Systems (LCCC) funded by the Swedish Research Council (VR), the Excellence Center Linköping - Lund in Information Technology (ELLIIT), and the Wallenberg Autonomous Systems Program (WASP).

References

[1] Ahmed Ali-Eldin, Maria Kihl, Johan Tordsson, and Erik Elmroth. Analysis and characterization of a video-on-demand service workload. In *Proceedings of the 6th ACM Multimedia Systems Conference*, pages 189–200. ACM, 2015.

[2] Meeyoung Cha, Pablo Rodriguez, Jon Crowcroft, Sue Moon, and Xavier Amatriain. Watching television over an ip network. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, pages 71–84. ACM, 2008.

[3] Manxing Du, Åke Arvidsson, Maria Kihl, and Christina Lagerstedt. Analysis of prefetching schemes for a tv-on-demand service. to be submitted to International Conference on Digital Communications 2015, 2015.

[4] Chai-Hien Gan, Phone Lin, and Chung-Min Chen. A novel prebuffering scheme for iptv service. *Computer Networks*, 53(11):1956–1966, 2009.

[5] Vijay Gopalakrishnan, Rittwik Jana, KK Ramakrishnan, Deborah F Swayne, and Vinay A Vaishampayan. Understanding couch potatoes: measurement and modeling of interactive usage of iptv at large scale. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 225–242. ACM, 2011.

[6] Jahon Koo and Kwangsue Chung. Adaptive channel control scheme to reduce channel zapping time of mobile iptv service. *Consumer Electronics, IEEE Transactions on*, 57(2):357–365, 2011.

[7] Chae Young Lee, Chang Ki Hong, and Kang Yong Lee. Reducing channel zapping time in iptv based on user's channel selection behaviors. *Broadcasting, IEEE Transactions on*, 56(3):321–330, Sept 2010.

[8] Hongliang Yu, Dongdong Zheng, Ben Y Zhao, and Weimin Zheng. Understanding user behavior in large-scale video-on-demand systems. In *ACM SIGOPS Operating Systems Review*, volume 40, number 4, pages 333–344. ACM, 2006.

[9] Huimin Zhang. User behaviour analysis and prediction methods for large-scale video-on-demand system. Master's thesis, Uppsala Universitet, 2015.