SocialTube: P2P-Assisted Video Sharing in Online Social Networks

Sherly Jose
PG Scholar
Maria College of Engineering and Technology, Attoor
Tamilnadu, India
Email: sherlyjose4@gmail.com

Abstract: Video sharing has been an increasingly popular application in online social networks. However, its sustainable development is severely hindered by the intrinsic limit of the client/server architecture deployed in current OSN video systems, which is not only costly in terms of server bandwidth and storage but also not scalable with the soaring amount of users and video content. The peer-assisted Video on Demand (VoD) technique, in which participating peers assist the server in delivering video content, has been proposed. Unfortunately, videos can only be disseminated through friends in OSNs. Therefore, current VoD works that explore clustering nodes with similar interests or close location for high performance are suboptimal, if not entirely inapplicable, in OSNs. Based on our long-term real-world measurement of over 1,000,000 users and 2,500 videos on Facebook, we propose SocialTube, a novel peer-assisted video sharing system that explores social relationship, interest similarity, and physical location between peers in OSNs. Specifically, SocialTube incorporates four algorithms: a social network (SN)-based P2P overlay construction algorithm, an SN-based chunk prefetching algorithm, a chunk delivery and scheduling algorithm, and a buffer management algorithm. Experimental results from a prototype on PlanetLab and an event-driven simulator show that SocialTube can improve the quality of user experience and system scalability over current P2P VoD techniques.

1 INTRODUCTION

Online social networks (OSNs) (e.g., Facebook, Twitter) are now among the most popular sites on the Web. An OSN provides a powerful means of establishing social connections and sharing, organizing, and finding content. For example, Facebook presently has over 500 million users. Unlike current file or video sharing systems (e.g., BitTorrent and YouTube), which are mainly organized around content, OSNs are organized around users. OSN users establish friendship relations with real-world friends or virtual friends, and post their profiles and content such as photos, videos, and notes to their personal pages.

In order to investigate the video watching behaviors of users in OSNs, we queried more than 1,000,000 users and retrieved about 2,500 public visible videos meta data on Facebook. Our measurement reveals that (1) most of the viewers of a user’s videos are the user’s close friends, (2) most video views are driven by social relationships, and the rest are driven by interests, and (3) viewers of the same video tend to reside in the same location. Based on our observations, we
propose Social Tube, a system that explores the social relationship, interest similarity and location to enhance the performance of video sharing in OSNs. Specifically, an OSN has a social network (SN)-based P2P overlay construction algorithm that clusters peers based on their social relationships and interests. Within each cluster, nodes are connected by virtue of their physical allocation in order to reduce video transmission latency. SocialTube also incorporates an ASN-based chunk prefetching algorithm to minimize video playback startup delay. We have conducted extensive experiments in an event-driven simulator and implemented a prototype on PlanetLab to evaluate the performance of Social Tube. Performance results show that SocialTube greatly reduces the workload of the server, improves the quality of playback, and scales well to a large client population. In the supplemental material, we present a chunk delivery and scheduling algorithm and a buffer management algorithm to further enhance the performance of SocialTube.

To our knowledge, this work is the first that studies the distinct characteristics of OSN video sharing that vary from other content-based system-wide video sharing, and builds a P2P-based video sharing system in an OSN by leveraging those characteristics for higher performance. Our previous conference version of this article [22] introduces the basic trace data analysis and design of SocialTube. This article presents more trace data analytical results. It also presents new SocialTube mechanisms including locality-aware video prefetching mechanism, two policies to increase the chunk delivery abilities, and buffer management algorithm. This article further presents more simulation results and the experimental results for the SocialTube prototype on the real-world PlanetLab test bed.

2 FACEBOOKMEASUREMENTANDANALYSIS

In this section, we present our Facebook trace measurement results and give an in-depth perspective of Facebook video viewing patterns, that shows the necessity of peer assistance in OSN video sharing and provides a direction for the design of a P2P video sharing system in OSNs. We used breadth-first search [23] to query over 1,000,000 users seeded by 5 users in the USA. In order to avoid overloading the Facebook, we sent a query to Facebook every 5 s. We can only see the video activities of the users who are friends or FOFs of the crawler and the users that chose “everyone” as their video access option. Because of this access limit, we only found about 2,500 videos and 12,000 users who watched these videos during the time period from Jul. 2007 to Aug. 2010, which is used as a sample for the video sharing and watching activities. The collected dataset includes the information about user friendship relations, interests, locations, and videos uploaded and shared by users. For each video, we retrieved the video metadata such as its title, length, and viewers when available. To respect the privacy of the users, we anonymized the user names before storing the data in our database. We only crawled the video metadata of these users, with other personal information untouched.

2.1 Popularity of Videos on Facebook

First, we investigate the popularity of videos on Facebook over the years. Number of videos corresponding to the time they are uploaded in our collected video pool. It shows that the number of videos uploaded to Facebook increases sharply along with time. Since Facebook launched video service in 2007, the increasing trend of video uploading has never slowed...
down, making it one of most popular applications on Facebook the video length Distribution in our collected video pool.

2.2 Effect of Social Distance on Video Viewing Patterns

Social distance between two users in the social network graph represents the closeness of their relationship. If two users are directly connected in the social network, their social distance is 1; if one user is a friend of another user’s friend, then the social distance between them is 2, and so on. Next, we investigate the impact of social distance on user video viewing patterns.

We call the viewers who have watched almost all videos of a user the user’s followers, and call other viewers non-followers. We use a threshold $T_h$ for the percentage of all the videos of a user that a viewer watches in order to become a follower, and set $T_h = 80\%$ in this analysis.

2.3 Effect of Interest on Video Viewing Patterns

Next, we explore the correlation between user interests and video viewing patterns. We selected a sample of 118 distinct users that watched more than one video from our dataset and manually classified the videos they watched into 19 interest groups based on video content. The 19 interest groups were determined based on the video categories in YouTube such as gaming, rock music, and action movies. For each user, we calculated the percentage of viewed videos of each interest group. Then, we ranked these 19 interest groups in descending order of the percentage values. We calculated the average percentage value of the 118 users for each interest group rank and show the result. We observe that, on average, 46% of videos a user watched are on his/her top 2 interests topics, 79% of videos a user watched are on his/her top 3 interests topics, and 94% are on his/her top 4 interests topics.

2.4 Effect of Physical Location on Video Viewing Patterns

We also analyze the geographical locations of users who view the same videos in order to see whether location can also be leveraged for video sharing in OSNs. In Facebook, some users input their current residence city in their profiles. Because many friend relationships in Facebook are connected by offline relationships such as classmates or colleagues, this produces a strong location clustering effect. This result conforms to the observation that most of the wall posts are present within the local physical region. This effect could make P2P video sharing systems in OSNs more efficient by enabling geographically close nodes to share videos between each other.

2.5 Active Life Period of Videos

We measured the percentage of views of a video in each month after the video is uploaded out of all views. We found that videos in Facebook have an active life period of about one month. Views in this period account for more than 90% of all views. After one month, there are only occasional views.
3 THE DESIGN OF SOCIAL TUBE

The Facebook measurement video sharing in Facebook is increasingly popular (O1) and may generate a heavy burden on the video server (O2). Fortunately, the length of time that the users stay online in Facebook is rapidly increasing (O3), enabling the possibility for P2P video sharing among the online users themselves. Therefore, P2P-assisted video sharing is a promising strategy in OSNs. Based on observations O4-O10 on the characteristics of video viewer behavior in Facebook, we propose SocialTube, a P2P video sharing system for OSNs.

We first introduce the basic concepts and strategies used in SocialTube. In Facebook, each node can upload a video to the Facebook video server or an external link to a video from an external server. In this paper, we use server to represent all video source servers, including both Facebook and external video servers. Similar to current peer-assisted content delivery mechanisms, the peers in SocialTube store videos they have watched previously for video redistribution. In SocialTube, a video is divided into small chunks with an fixed size.

In current video sharing in Facebook, a node always requests the server for videos uploaded by source nodes. We let the server keep track of the video watching activities of viewers of a specific source node in order to identify and update its followers and non-followers based on SocialTube’s predefined threshold of $T_l$ and $T_h$. This duty can be assigned to the source node itself if it has sufficient capacity. The nodes in the system will periodically report their video watching activities to the server. When the server determines that a peer is a follower of the source node, it notifies the source node, which notifies all nodes in its swarms about the follower. Consequently, the follower becomes a member of each of the swarms, and all swarm peers in each swarm connect to it. When the server determines that a peer is a non-follower of the source node, the server determines its interests based on the contents of videos the peer visited, and notifies the source node about the non-follower along with its interests. The source node then notifies the peers in the clusters of the interests of that non-follower, and notifies the non-follower about the clusters. Then non-followers connect to all followers and the source and to a few physically close nodes in each cluster.

To reduce the video startup latency, we propose a push-based video prefetching mechanism in SocialTube. In SocialTube, when a source node uploads a new video to the server, it also pushes the prefix (i.e., first chunk) of the video to its followers and to the interest clusters matching the content of the video. The prefix receiver stores the prefix in their cache. Those interest cluster-peers and followers who are not online when the source node pushes the prefix will automatically receive it from the source node or the server once they come online. After the source node leaves, the responsibility to push the prefix falls to the server. Since these followers and interest cluster-peers are very likely to watch the video, the cached prefixes have a high probability of being used.

Once the nodes request the videos, the locally stored prefix can be played immediately without delay. Meanwhile, the node tries to retrieve the remaining video chunks from its swarm peers. Similar to BitTorrent, SocialTube allows a request to request 4 online nodes at the same time to provide the video content in order to guarantee provider availability and achieve low delay by retrieving chunks in parallel. It first contacts interest cluster-peers, then followers, then the source...
node. If the requester still cannot find 4 providers after the source node is contacted, it resorts to the server as the only provider. Considering the high capacity of the server, the requester does not need to have 4 providers if it has the server as a provider. This querying order can distribute the load of chunk delivery among the swarm peers while providing high chunk availability. The algorithm takes advantage of all resources for efficient video sharing without overloading specific nodes. The server can guarantee the availability of the video, even if the number of online users in a swarm is small.

In this experiment, we randomly selected 5 per-node overlays and chose a client in each overlay. We let each source node have 30 videos in 5 interests. A client initially has interests. In every simulation cycle, each client receives a prefix from its source node and chooses [1-4] videos to watch. If the client chooses a video not in its current interests, the video’s interest is added to the client’s interest list and it joins in the interest-cluster of this interest and receives the pushed prefix of videos in this interest in Social Tube and NetTube. The average value of the prefetching accuracy of the 5 clients versus the number of watched videos of a client over time. The figure illustrates that as the number of watched videos increases, the prefetching accuracy of SocialTube and NetTube increases. This is because in SocialTube and NetTube, a client watches more videos, it joins in more interest-clusters and receives more prefixes, and thus its prefetching accuracy increases. Again, NetTube generates slower prefetching accuracy than SocialTube due to the same reason. The prefetching accuracy of PA-VoD and random is not affected by the number of watched videos as they don’t have interest clusters. Also, they produce low prefetching accuracy due to the same reasons as explained before.

4. CONCLUSION

Video sharing is an increasingly popular application in OSNs. However, the client/server architecture deployed by current video sharing systems in OSNs costs a large amount of resources (i.e., money, server storage) for the service provider and lacks scalability. Meanwhile, because of the privacy constraints in OSNs, the current peer-assisted Video-on-Demand (VoD) techniques are suboptimal if not entirely applicable to the video sharing in OSNs. In this paper, we crawled video watching trace data in one of the largest online social network websites Facebook, from Jul. 2007 to Aug. 2010 and explored the users’ video viewing patterns. We found that in a user’s viewer group, 25% of viewers watched all videos of the user driven by social relationship, and the viewing pattern of the remaining nodes is driven by interest. Based on the observed social and interest relationship in video watching activities, we propose SocialTube, which provides efficient P2P-assisted video sharing services. Extensive simulation results show that SocialTube can provide a low video startup delay and low server traffic demand. We also implemented a prototype in PlanetLab to evaluate the performance of SocialTube. The experimental results from the prototype further confirm the efficiency of SocialTube.
REFERENCES


ABOUT THE AUTHOR

Sherly Jose, Pursuing Master of Engineering in Computer Science and Engineering, Maria College of Engineering and Technology, Attoor, Anna University Chennai.