Summarizing User-Contributed Comments

Elham Khabiri and **James Caverlee** and **Chiao-Fang Hsu**
Texas A&M University
College Station, TX

Abstract

User-contributed comments are one of the hallmarks of the Social Web, widely adopted across social media sites and mainstream news providers alike. While comments encourage higher-levels of user engagement with online media, their wide success places new burdens on users to process and assimilate the perspectives of a huge number of user-contributed perspectives. Toward overcoming this problem we study in this paper the comment summarization problem: for a set of n user-contributed comments associated with an online resource, select the best top-k comments for summarization. In this paper we propose (i) a clustering-based approach for identifying correlated groups of comments; and (ii) a precedence-based ranking framework for automatically selecting informative user-contributed comments. We find that in combination, these two salient features yield promising results. Concretely, we evaluate the proposed comment summarization algorithm over a collection of YouTube videos and their associated comments, and we find good performance in comparison with traditional document summarization approaches (e.g., LexRank, MEAD).

Introduction

Participatory information environments are growing in popular interest – from Web 2.0 social news aggregators to digital libraries incorporating social computing features to enterprise social networks. Across these varying environments, one of the key factors driving the popularity and "stickiness" of these services is their emphasis on *user-driven commenting and discussion*. By encouraging users to comment, resources in these systems (like videos, images, news articles) can become "social" resources that reflect the attitudes and interests of the community of users in a way that may depart from the viewpoint of system experts, editors, and the content of the underlying information resource itself. Popular websites like NYtimes.com, Digg.com, CNN.com, as well as a host of weblogs, collectively manage millions of user-contributed comments on news articles, images, videos, and so forth. For any particular web resource, however, it is challenging to quickly ascertain the overall themes and thrusts of the mass of user-contributed comments. While some users may be interested in scanning over hundreds or thousands of comments, there has been a shift in recent years towards providing guidance to users to focus their attention on particular comments.

- Editorial selection: One approach is to rely on human editors to select representative comments. This is the approach taken by NYTimes which provides comment highlights: "A selection of the most interesting and thoughtful comments that represent a range of views." Editorial comments, however, may be biased toward the particular worldview of the comment selector and not representative of the themes of the comments themselves.
- Collaborative recommendation: In a separate direction, several sites allow users themselves to recommend comments (e.g., through a thumbs-up/thumbs-down rating mechanism). For example Digg.com offers users the option of sorting comments by the number of community votes to prioritize the comments. Collaborative recommendations, while beneficial for aggregating a community's perspective, may not be very informative, favoring funny comments or the comments that are submitted by popular users. In addition, collaborative recommendations require adequate participation rates to be successful.
- Keyword Cloud: Rather than select particular comments, many blogs support a keyword-based word cloud to show the most frequent topics and keywords used. For example, streamhacker.com, a blog about platforms, libraries, and languages, uses a tag cloud for each post. While keyword-based summaries may convey the overall flavor of a group of comments, the keywords themselves lack the context and structure of a sentence-based comments for more detailed understanding.

With these issues in mind, we propose to automatically summarize user-contributed comments through a process of identifying and extracting key informative comments. This approach is inspired by recent efforts at automatic text summarization for creating a compact version of either a single document or a collection of documents (Radev 2004), (Mihalcea and Ceylan 2007). Concretely, we propose (i) a clustering-based approach for identifying correlated groups of comments; and (ii) a precedence-based ranking framework for automatically selecting informative user-

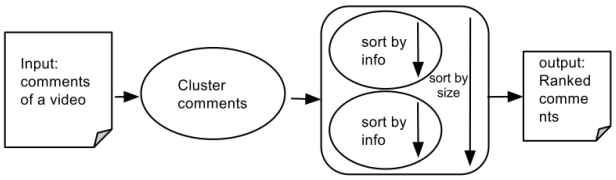

Figure 1: The overall comment summarization approach

contributed comments. The approach takes as a parameter k, for selecting the best top-k comments for summarization. Our intuition is that comments may belong to one of several overall themes within a collection of comments. By identifying significant comments within each cluster we may promote both informative and representative comment summaries. To study comment summarization, we rely on the YouTube video sharing site and a collection of 30 videos for which we have sampled comments. We test the quality of top-k comments selected algorithmically versus a gold set based on a five-subject user study, in which each user has selected at least five comments deemed informative and representative of the entire collection of comments. We find good performance in comparison with traditional document summarization approaches.

Approach

Our overall goal is to select the most representative comments with respect to a resource from a large collection of user-contributed comments. At the same time the selected comments should cover different viewpoints about the associated resource that can highlight various aspects of the resource. We define V as the set of all resources that we have in our dataset $V = \{v_1, v_2, ..., v_n\}$. Each resource v_i is associated with a set of comments $C_i = \{c_1, c_2, ..., c_m\}$, where each c_j is a single comment that we consider as a bag of words. Here m is the total number of comments. Our goal is to extract a subset of C_i, $S_{C_i} \subset C_i$, that are the k most representative comments: $S_{C_i} = \{s_1, s_2, ..., s_k\}$, based on a ranking of all of the comments associated with a resource and where k is a tunable parameter. Since our goal is to summarize a large set of comments for quick understanding, we will typically require $k \leq 5$, though larger values may be appropriate in some situations.

Our overall approach is to (i) identify groups of thematically-related comments through an application of traditional clustering, (ii) rank clusters according to a measure of significance, (iii) then rank comments within each cluster according to a measure of importance (as illustrated in Figure 1). To identify important and informative comments within a cluster we explore two approaches: a term-importance approach that rewards comments with "important" terms and a precedence-based approach that rewards comments based on a PageRank-style random walk over a comment graph.

Identify Groups of Related Comments

To identify groups of related comments, we have investigated both k-means clustering and topic-based clustering based on Latent Dirichlet Allocation (Blei, Ng, and Jordan 2003). We focus our presentation here on the topic-based approach.

Topic Clustering

Latent Dirichlet Allocation (LDA), is a generative model that can be used to identify the underlying topics that documents are generated from. We use LDA to extract T topics out of the comments associated with a single resource. That is, we have a set of comment "documents" $D = \{d_1, d_2, ..., d_n\}$ and a number of topics $T = \{t_1, ..., t_m\}$. Any document d_i can be viewed by its topic distribution. For example $Pr(d_1 \in t_1) = 0.70$ and $Pr(d_1 \in t_2) = 0.20$ and so on. We modify the original soft clustering of LDA to a hard clustering by considering each comment as belonging to a single topic (cluster) $r = argmax_r Pr(t_r|c) = argmax_r Pr(c|t_r)Pr(t_r)$, where r is the topic number that has the maximum likelihood for each comment. Hence, the output of the LDA-based topic clustering approach is an assignment from each comment to a cluster.

Identifying Significant In-Cluster Comments

After producing our clusters, the next step is to select the most informative comments in each of them. Users want to focus immediately on a handful of key comments that communicate the key ideas from across all comments. We need some way of selecting one or a handful of comments per cluster. That is, given a cluster, select a comment (or a few) that best expresses the cluster. We consider two approaches: a term-importance based and a precedence-based approach.

Term Importance

The first approach to ranking comments within a cluster is by awarding more points to comments containing "important" terms. The intuition is that comments containing more significant terms are themselves more significant. We consider two approaches to term importance: a vector space (geometric) measure and an information theoretic measure of term importance. In selecting comments by vector space-based importance, $tf_{i,j}$ is defined as the number of time a $term_i$ appears in the comments of a particular $resource_j$ normalized by the total number of terms in the comments of that resource, and idf_i is the logarithm of total number of resources $|D|$ divided by the number of resources that $term_i$ appeared in. The importance of each comment c_k is the average of the importance of terms used in that comment using $tf\text{-}idf$ metric. In selecting comments by information theoretic importance, we measure how much information the presence or absence of a term contributes to the term appearing in the appropriate cluster. For each term of comment i in cluster k, $c_{i,k}$, we calculate the Mutual Information (MI) of

that term.

$$MI = \frac{N_{11}}{N}\log_2\frac{N_{11}N}{N_{1.}N_{.1}} + \frac{N_{10}}{N}\log_2\frac{N_{10}N}{N_{1.}N_{.0}} + \frac{N_{01}}{N}\log_2\frac{N_{01}N}{N_{0.}N_{.1}} + \frac{N_{00}}{N}\log_2\frac{N_{00}N}{N_{0.}N_{.0}}$$

The first subscript indicates if a comment contains the term or not and the second subscript shows if we are considering the comments in the current cluster or other clusters. N is the total number of comments. Suppose that we want to find the MI of the term t of a resource which is grouped in cluster k. N_{10} shows the number of the comments that contain term t and are not in cluster k. Finally we rank the comments by the average MI of all the terms in each comment: $MI(c_{i,k}) = \frac{\sum_{t \in c_{i,k}} MI(t;k)}{|w|}$.

Precedence-based Ranking

In recent years, graph-based ranking methods, including TextRank (Mihalcea) and LexRank (Radev 2004) have been proposed for document summarization. Similar to Google's PageRank algorithm (Page et al. 1998) or Kleinberg's HITS algorithm (Kleinberg 1999), these methods first build a graph based on the similarity relationships among the sentences in a document and then the importance of a sentence is determined by a random walk over the sentence graph. Similarly, we are interested to explore whether a random walk over a comment graph can reveal important comments. Our hypothesis is that comments that reference an earlier comment may confer some level of implicit endorsement on the earlier comment, in essence echoing the ideas of the earlier comment. Comments that are never echoed are less significant in the aggregate, whereas comments that insert new ideas that are repeated by others should be rewarded. Concretely, we consider a link between two comment nodes in the comment graph if the more recent node (comment) contains m terms that also occur in the earlier node. Performing a random walk we can find comments with more "votes of support" from the later comments. To calculate the score of a comment $S(c_i)$ we add the score of all the neighbors pointing to it divided by the number of output links of each of these neighbors. We used 0.85 as our damping factor d. $S(c_i) = d \times \sum_{c_j \in neighbor(c_i)} S(c_j)/count(c_j) + (1-d)$. The random walk continues iteratively until the scores of the nodes converge.

Experiments

In this section, we present an experimental study of comment summarization over a collection of YouTube web videos and their associated comments.

User-based Evaluation

In this paper we evaluate different algorithms based on a user study we conducted on 5 subjects and 30 videos. The selected videos received between 500 and 1,000 comments each. To make evaluation possible we selected the first 50 comments associated with each video and showed them to our subjects. We asked each subject to mark the comments that they find interesting and informative. Aggregating the number of times each comment is selected, each comment receives a score from 0 to 5. A score of 5 means all of the subjects found the comment informative and interesting, whereas a score of 0 means that none of the subjects did. We used the well known method normalized discounted cumulative gain (NDCG) that values highly relevant comments that have appeared earlier in the ranking result. The ideal ranking is gained by user relevance score which is the average score of the subjects for each video.

Dataset

We crawled 17,600 videos from the YouTube website using Tubekit (Shah 2008). To sample videos, we issued queries drawn from two different policies: (1) Random word selection from an English dictionary resulting in 240 queries; and (2) The top most popular queries based on Google trends from September to November 2009, resulting in 3,596 queries.

Cluster-based Ranking Evaluation

First we study which of the two clustering methods – k-means or LDA-based topic clustering – is more suitable for user-contributed comments. Measures of qualities for our clusters are *Cohesion* and *Separation*. *Cohesion* measures how similar the comments in one cluster are to each other. On the other hand *Separation* measures how dissimilar are the comments across clusters. We apply these two measures to the clusters of comments of all the videos in our dataset.

We found that topic-based clustering gives us higher separation and cohesion in comparison with k-means clustering. Details are omitted due to the space restriction. Therefore, in the rest of the paper we use topic-based clustering for grouping thematically-related groups of comments.

To Cluster or not to Cluster Here we want to see if clustering the comments can make any difference in the quality of comments selected as summary comments. That is, does thematically grouping comments lead to better coverage of interesting comments? We compare the NDCG of the ranking result for the case that uses clustering (with a simple tf-idf approach for ranking comments within a cluster) and the case that uses only a basic tf-idf measure for ranking comments by their average tf-idf score (and ignoring any cluster or thematic structure). We found that the no-cluster approach generally gives better results, perhaps revealing that cluster structure is of little value for comment summarization.

However, we find that when we combine precedence-based ranking for selecting high-quality comments from within a cluster, that the cluster-based approach results in a higher NDCG relative to the no-cluster case, as shown in Figure 2. This result suggests that in combination, these two methods emphasize on valuable and diverse comments (that cross multiple thematic groups of comments).

In-cluster Ranking Evaluation

Now that we see topic clustering will reveal other aspects of organizing comments, we study which of the in-cluster

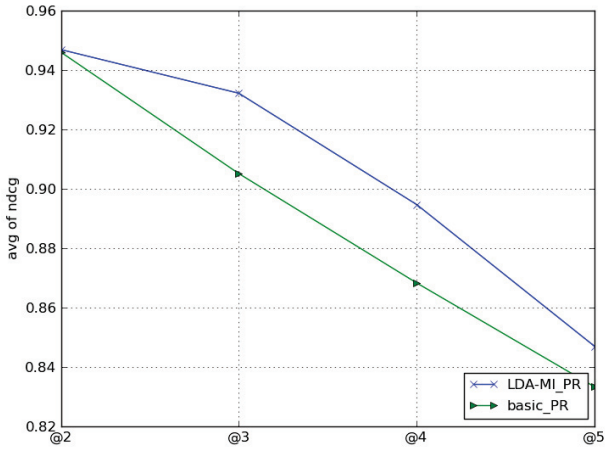

Figure 2: Compare cluster-based ranking vs. basic tf-idf based ranking in combination with precedence-based ranking

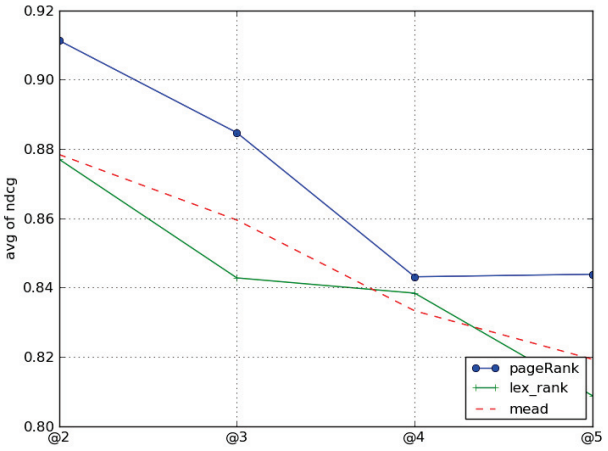

Figure 3: Compare precedence-based ranking versus MEAD and LexRank

ranking methods are more successful. Comparing the information theory-based method (MI) with vector space (geometric) based importance (tf-idf) when it is applied inside each of our result clusters, shows that MI has better performance than tf-idf in selecting the most representative comments. This can be justified by MI's focus on terms that contribute the most to a particular cluster.

Precedence-based Ranking Evaluation

Finally, we compare the precedence-based rank approach with two well-known traditional document summarization approaches: MEAD and LexRank (Radev 2001; 2004). These are graph-based ranking methods that first build a graph based on the similarity relationships among the sentences in a document and then the importance of a sentence is determined by taking into account the global information of the graph recursively. Much summarization research (Mihalcea ; Shen et al. 2007; Mihalcea and Ceylan 2007) uses these two algorithms as the base for their customized ranking models and these two methods have shown good results for single and multi-document summarization.

Figure 3 compares our precedence-based ranking with these two methods and shows that our proposed method results in a higher NDCG when considering 2, 3, 4, and 5 comments. These results suggest that massively generated user comments may require new summarization methods that emphasize on the unique properties of user comments compared to traditional assumptions of document summarization (like the importance of comment order, text references to earlier comments, and so on).

Conclusion

In this paper we have studied the comment summarization problem over a collection of YouTube videos. Our approach includes clustering the comments and selecting the most representative comments of each cluster. We also proposed a precedence-based ranking method that in combination with topic-based clustering yields overall higher performance in comparison with traditional document summarization approaches.

References

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.* 3:993–1022.

Kleinberg, J. M. 1999. Hubs, authorities, and communities. *ACM Comput. Surv.* 31(4es).

Mihalcea, R., and Ceylan, H. 2007. Explorations in Automatic Book Summarization. In *EMNLP-CoNLL*, 380–389.

Mihalcea, R. Language independent extractive summarization. In *Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*, 49–52. Stroudsburg, PA, USA: Association for Computational Linguistics.

Page, L.; Brin, S.; Motwani, R.; and Winograd, T. 1998. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project.

Radev, D. R. 2001. Experiments in single and multidocument summarization using mead. In *In First Document Understanding Conference*.

Radev, D. R. 2004. Lexrank: graph-based centrality as salience in text summarization. *Journal of Artificial Intelligence Research* 22.

Shah, C. 2008. Tubekit: a query-based youtube crawling toolkit. In *JCDL*, 433.

Shen, D.; Sun, J.-T.; Li, H.; Yang, Q.; and Chen, Z. 2007. Document summarization using conditional random fields. In *Proceedings of the 20th international joint conference on Artifical intelligence*, 2862–2867. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.