Up Next: Retrieval Methods for Large Scale Related Video Suggestion

Michael Bendersky
Google, Inc.
bemike@google.com

Lluis Garcia-Pueyo
Google, Inc.
lgpueyo@google.com

Jeremiah Harmsen
Google, Inc.
jeremiah@google.com

Vanja Josifovski
Google, Inc.
vanjaj@google.com

Dima Lepikhin
Google, Inc.
lepikhin@google.com

ABSTRACT

The explosive growth in sharing and consumption of the video content on the web creates a unique opportunity for scientific advances in video retrieval, recommendation and discovery. In this paper, we focus on the task of video suggestion, commonly found in many online applications. The current state-of-the-art video suggestion techniques are based on the collaborative filtering analysis, and suggest videos that are likely to be co-viewed with the watched video. In this paper, we propose augmenting the collaborative filtering analysis with the topical representation of the video content to suggest related videos. We propose two novel methods for topical video representation. The first method uses information retrieval heuristics such as tf-idf, while the second method learns the optimal topical representations based on the implicit user feedback available in the online scenario. We conduct a large scale live experiment on YouTube traffic, and demonstrate that augmenting collaborative filtering with topical representations significantly improves the quality of the related video suggestions in a live setting, especially for categories with fresh and topically-rich video content such as news videos. In addition, we show that employing user feedback for learning the optimal topical video representations can increase the user engagement by more than 80% over the standard information retrieval representation, when compared to the collaborative filtering baseline.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Experimentation

KDD'14, August 24–27, 2014, New York, NY, USA.

http://dx.doi.org/10.1145/2623330.2623344.

Keywords

Video retrieval, related video suggestion, video representation

1. INTRODUCTION

The World Wide Web project was originally conceived as a means to publish, share and retrieve textual information and did not aim *"to do research into fancy multimedia facilities such as sound and video"* [1]. However, with the growth of the web, video and other multimedia formats became an increasingly large and important part of it. The recent proliferation of mobile devices that enable instant video capturing, sharing and consumption, further accelerated this trend.

As an example, at the time of the writing of this paper, over 6 billion hours of video are watched monthly on YouTube, the most popular video-sharing website, an increase of 50% compared to last year. Furthermore, 100 hours of new video content are uploaded to YouTube every minute [2].

These statistics demonstrate that there is an ever-growing abundance of video content on the web. This content ranges from six-second long mobile video clips on Vine and user-uploaded footage on YouTube, to news stories on news websites and TV shows and movies on subscription-based streaming services such as Hulu and Netflix.

This abundance of content supply and demand has brought about an urgent need for increasingly sophisticated techniques for video retrieval, recommendation and discovery. In particular, in this paper, we focus on the task of *video suggestion*, which is common in many online applications.

We define *video suggestions* as a ranked list of *related videos* shown to the user in response to the video that she is currently viewing (which we refer to as the *watch video*). Video suggestions are an important part of the experience of watching video on the web today, and they are steadily becoming prevalent on more and more video websites. As an illustration, Figure 1 showcases three major use cases for video suggestions, including suggestion of (a) news stories on CNN, (b) TV show episodes on Hulu, and (c) related videos on YouTube.

As shown in previous work, successful deployment of video suggestion algorithms can result in a significant positive ef-

[1] As stated in the original HyperText proposal: `http://www.w3.org/Proposal.html`.

Figure 1: Examples of video suggestions on the web, showcasing suggestions based on the currently watched video on CNN, Hulu and YouTube websites.

fect on user engagement and user experience [7, 13]. For instance, Davidson et al. [13] show that co-visitation based video suggestions more than double the click-through rate, compared to a baseline that does not take the watch video into account.

Video suggestions can be either based on the watch video alone [7], or can be personalized by incorporating information about the user (e.g., search history [11]). Current state-of-the-art video suggestion systems are based on the collaborative filtering analysis. Most generally, videos suggested by these systems are likely to be co-viewed with the watch video by the majority of the users [7, 11, 13].

While this approach works well for popular and often watched videos, it is less applicable to fresh videos or tail videos with few views, since they have very sparse and noisy co-view data. For these videos, collaborative filtering analysis based solely on co-views may yield either low-quality suggestions or no suggestions at all.

To address the problem of tail content in other domains such as recommendations based on textual information, researchers developed hybrid approaches [10] that combine information about the item content with the collaborative filtering information to improve recommendation. While there have been some small scale studies on hybrid recommendation systems for video suggestion [31], this paper, to the best of our knowledge, is the first published research study of a large scale deployment of such a system.

To achieve the goal of content-based video representation, we first model a video as a set of topics that are mined from a variety of sources including the video metadata, frequently used uploader keywords, common search queries, playlist names, Freebase entities and Wikipedia articles. These topics are then used to retrieve related videos for suggestion.

Thus, our approach to content-based video suggestion is akin to the standard information retrieval scenario. The topical representation of the watch video is the *query* that is issued to the inverted index that stores the topical video representations as *documents*. Since we cast the video suggestion problem as a retrieval over an inverted index, we can use query optimization algorithms [8, 14] to efficiently find the optimal video suggestions even in a very large corpus.

The highest ranked documents retrieved in response to the query are returned as content-based video suggestions. These suggestions can be either directly shown to the user, or used to augment the output of the co-view based video suggestion system.

In order to achieve an effective topical video representation, we assign weights to each topic associated with a video. We propose two techniques for topical video representation.

The first technique is based on the well known information retrieval heuristics such as computing topic frequencies and topic inverse document frequencies. The second technique leverages the implicit user feedback (such as video co-watches) available in the online setting. This feedback is used for a supervised learning of the optimal topic weights.

We empirically evaluate these two topical representation methods. We find that both of these representations have a significantly positive effect on the quality of video suggestions. Furthermore, our evaluation demonstrates that learning topic weights from user feedback can increase the user engagement (compared to the collaborative filtering approach) by more than 80% over the standard information retrieval representation.

To evaluate our approaches, we conduct a large scale live experiment on millions of YouTube video. The live experiment demonstrates that a hybrid video suggestion system that incorporates topic-based retrieval significantly outperforms a purely co-view based suggestion system. These improvements are especially high for fresh videos and videos with rich topical representations.

There are several important contributions in this paper. First, we formulate the video suggestion task as an information retrieval problem and demonstrate that this formulation enables effective and efficient deployment of video suggestion on a very large scale. Previous work on video retrieval was mainly performed offline on relatively small collections such as TRECVID collections [20], or small pre-selected samples of online videos [31]. To the best of our knowledge, this is the first published study to address the challenges of real-time video retrieval using topical representation in a large scale web collection.

Second, we demonstrate that the richness of the implicit user feedback available in the online setting can be leveraged to improve the effectiveness of topical video representations. To this end we employ a novel learning technique that derives the optimal topic weights from co-view information.

Third, we describe the architecture of the hybrid video suggestion system that combines the collaborative filtering information with the topic-based video information.

Finally, we thoroughly evaluate the performance of our system using both user simulation and a large scale live experiment. This evaluation demonstrates the superiority of

the proposed hybrid video suggestion system to the current state-of-the-art collaborative filtering approaches.

The rest of the paper is organized as follows. Section 2 describes video representations using topics. Section 3 describes retrieval using topics weighted by information retrieval heuristics, and Section 4 describes retrieval using topic transitions learned from user watch behavior. Section 5 provides a holistic view of the proposed video suggestion system. In Section 6 we overview the related work. Section 7 desribes the evaluation of our techniques using both user simulation and a large scale live experiment. Section 8 concludes the paper.

2. VIDEO REPRESENTATION

In this section, we discuss how a video is represented via a set of associated *topics*. These topics serve as a semantic representation of the video content, and are used by the retrieval algorithms in the next sections. The topics are derived from a variety of sources that can be associated with a video such as video metadata, uploader keywords, common search queries, playlist names, etc.

For the purpose of this paper, we use the notation V both to refer to the video itself, and the set of topics associated with it, interchangeably. Therefore, the notation $\tau \in V$ is used to state that video V is annotated with topic τ.

2.1 Topic Assignment

There is a large variety of methods described in the scientific literature (e.g., [16, 27, 28, 29, 30] among many others) on how to derive a set of annotations for images and videos. Some of these methods involve image and video analysis, while others rely on textual annotations, anchor text information and query logs.

In this paper, we follow the latter approach. This is due to the fact that there is a plethora of sources that can be utilized to annotate online videos with semantic information, including the video metadata, frequently used uploader keywords, common search queries, playlist names, and even sources from different domains such as Freebase entities and Wikipedia articles. The interested reader may refer to the YouTube official blog [1, 3] or work by Simonet [25] for more information about the YouTube video topic assignment process.

For the purpose of the retrieval algorithms discussed in the next sections, we will assume that we can reliably annotate a given video V with a set of topics using the sources described above [1, 3]. These topics represent the different semantic aspects of the video and map the video representation V into some existing concept space (e.g., Freebase concepts as shown in [3]).

As an example, Figure 2 demonstrates a movie trailer *World War Z* along with the topics associated with it and the topic weights. As can be seen in Figure 2, these topics capture several aspects of the video content including movie title, genre and leading actor, as well as information about the video itself.

In the rest of this paper, we will assume that the topical annotation can serve as a faithful representation of the video content. Therefore, we will use these annotations to retrieve videos related to the *watch video* – video that is currently being viewed by the user.

For this purpose, we will represent both the watch video and the potentially related videos as vectors of topic weights,

Figure 2: Example of topics associated with a video, and their corresponding weights.

and compute a vector dot product to determine a similarity of a <*watch video, related video*> pair. Related videos will then be ranked by their similarity score to the watch video.

2.2 Topics Versus Co-View Graph

It is important to note that the topical video representation described in the previous section has very different characteristics from the *co-view* video representation, used in previous work on related video discovery [7]. In the co-view video representation, each video is represented by a node in the co-view graph. A node is then linked to the other nodes in the graph if it is often viewed with them in the same session [7]. In this approach, related video suggestion is done based on the nodes that are in the proximity to the watch video in the co-view graph.

Therefore, in the co-view approach two videos are potentially related if and only if they have a strong connection in the co-view graph, i.e., they were often watched in the same session. This approach works well for popular videos with many views and high node connectivity. However, it is much less reliable for videos with little or no views. For these videos, using the co-view approach may lead to spurious results, or yield no candidates for suggestions.

In contrast, the topical video representation does not require an explicit co-view information to deem two video related. Instead, if two videos share (some of) the same topics they will be related, even if they were never watched in the same session before.

This semantic approach enables discovery of fresh, diverse and relevant content. It has been shown that implicit user feedback is often influenced by presentation bias [32], and click metrics do not fully correlate with relevance [21]. Topic-based video suggestion can therefore limit the *rich get richer* effect that can potentially arise when using solely the co-view information and disregarding the video content.

2.3 Topic Indexing

Video representation using topic weight vectors is akin to the *bag-of-words* document representation often used in the information retrieval applications. Therefore, we index the topic video representations in a standard inverted index structure [19] for efficient retrieval. Each video is represented using a topic weight vector, and indexed as an entry in the posting lists of its corresponding topics.

The inverted index structure enables efficient scoring of the related candidate videos in response to a watch video. Since we are only interested in a limited number of highest scoring related videos for a given watch video, we employ the WeakAnd query optimization technique first proposed by Broder et al. [8]. WeakAnd query optimization fully scores only a small fraction of the videos with score upper bound greater than a given threshold. This threshold is revised at query runtime as more videos are scored. To avoid fully scoring the document, the WeakAnd optimization maintains an upper bound of a document score, based on the maximum weight in each of the posting lists evaluated in response to the query. For more detailed description of the WeakAnd algorithm see [8, 14].

3. RETRIEVAL WITH WEIGHTED TOPICS

In Section 2.3 we described the topic indexing process. In this section, we use this topic index to develop a related video suggestion algorithm that is based on the standard information retrieval practices.

Recall that the videos in our system are represented as vectors of topic weights. Given this representation, for a watch video and related video pair $\langle V_W, V_R \rangle$ we derive the following score

$$sc(V_W, V_R) = q(V_R) \sum_{\tau \in V_W \cap V_R} \mathcal{I}_s(\tau) \frac{c(\tau, V_W)}{\log(1 + df(\tau))} c(\tau, V_R). \quad (1)$$

The score in Equation 1 has several components that are based on standard information retrieval practices.

First, the topic count function $c(\tau, V)$ returns a normalized count of videos that are annotated with the topic τ and are co-viewed with the video V. The topic count function estimates the *topicality* of video V with respect to topic τ.

Second, $\log(1 + df(\tau))$ is an inverse document frequency component that penalizes frequently occuring topics. Inverse document frequency demotes overly broad, vague and non specific topics, similarly to the *idf* term weighting in information retrieval applications.

Third, Equation 1 includes an indicator function

$$\mathcal{I}_s(\tau) = \begin{cases} 1 & df(\tau) < df_{max} \\ 0 & \text{else,} \end{cases}$$

where df_{max} is set to some large constant. The indicator function $\mathcal{I}_s(\tau)$ is inspired by the stopword removal in information retrieval applications, which removes very frequent stopwords at either indexing time or query time. We found that such stopword removal technique is useful for disregarding noisy and redundant topic matches from score computation, and improves both efficiency and effectiveness at query time.

Finally, Equation 1 takes into account the overall quality of the related video $q(V_R)$. Function $q(V_R)$ is based, among other factors on the video age, uploader, "thumbs up" and "thumbs down" counts, video popularity and freshness.

Note that since all the videos have roughly the same (small) number of topics associated with them, we do not apply any document length normalization method such as cosine similarity or pivoted length normalization [26]. Instead we simply use an unnormalized vector dot product as a scoring function in Equation 1.

4. LEARNING TOPIC TRANSITIONS

In Section 3 we described a retrieval approach that assigns topic weights based on a set of information retrieval heuristics (topic counts, inverse document frequency and stopword removal). In this section, we show that given the richness of the implicit user feedback available in the online setting, it is possible to learn the optimal transitions between the topics in the watch and the related videos. These transitions can then be directly leveraged for related video retrieval.

Consider, for instance, the example of the trailer for the *World War Z* movie shown in Figure 2. Assuming that the topic weights in Figure 2 are derived from the normalized co-occurence counts in Equation 1, the topic *Horror_Movie* is weighted lower than the topic *World_War_Z*. However, the *Horror_Movie* topic might be important for finding other related videos such as a trailer for the *Jurassic Park IV* movie. Therefore, a topic can have relatively low co-occurence counts for both the watch and the related videos, yet still be beneficial for the retrieval of relevant related videos.

Based on this intuition, we propose a novel machine learning approach that takes into account topic interactions and learns topic transition weights based on implicit user feedback. Our approach is based on pairwise classification of *<watch video, related video>* pairs.

In Sections 4.1 and 4.2 we describe how the topic transitions are represented. Then, in Section 4.3 we describe the optimization of the topic transition weights. Finally, in Section 4.4 we describe related video retrieval using topic transition weights.

4.1 Topic Transitions

Suppose that two potentially related videos are suggested to the user in response to the watch video V_W. One of the videos, $V_R^{(+)}$, was clicked and viewed by the user. The other video, $V_R^{(-)}$, was ignored by the user.

Most generally, we seek topic transition weight assignments such that the suggested related video $V_R^{(+)}$ will be preferred by the model to the suggested video $V_R^{(-)}$. This problem formulation is inspired by the pairwise classification approach which is common in learning-to-rank applications (e.g., see Burges et al. [9] or Joachims [17]).

Formally, we represent a potentially related video V_R that was suggested in response to a watch video V_W using a binary feature vector

$$\mathbf{x}_{V_R} = [\mathcal{I}_{V_R}(\tau_W, \tau_R) \colon \tau_W \in \mathbf{T}, \tau_R \in \mathbf{T}],$$

where $\mathbf{T}$ is the lexicon of all available topics, and $\mathcal{I}_{V_R}$ is a transition function from topic τ_W to τ_R such that

$$\mathcal{I}_{V_R}(\tau_W, \tau_R) = \begin{cases} 1 & \tau_W \in V_W, \tau_R \in V_R \\ 0 & \text{else} \end{cases}$$

Given a pair of related videos

$$P_R = \langle V_R^{(+)}, V_R^{(-)} \rangle,$$

we represent the pair using a ternary feature vector:

$$\mathbf{x}_{P_R} = [\mathcal{I}_{P_R}(\tau_W, \tau_R) \colon \tau_W \in \mathbf{T}, \tau_R \in \mathbf{T}],$$

where the transition function $\mathcal{I}_{P_R}(\tau_W, \tau_R)$ is defined such that

$$\mathcal{I}_{P_R}(\tau_W, \tau_R) = \mathcal{I}_{V_R^{(+)}}(\tau_W, \tau_R) - \mathcal{I}_{V_R^{(-)}}(\tau_W, \tau_R) \quad (2)$$

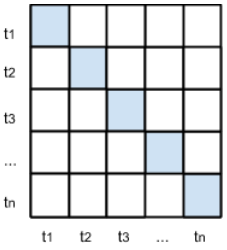


Figure 3: Topic transition matrix. Diagonal topic transitions are marked in grey.

It is easy to show that following the formulation in Equation 2, transition function $\mathcal{I}_{P_R}$ can take three values:

1. $\mathcal{I}_{P_R}(\tau_W,\tau_R) = +1$ iff τ_R is associated only with the positive example $V_R^{(+)}$.
2. $\mathcal{I}_{P_R}(\tau_W,\tau_R) = -1$ iff τ_R is associated only with the negative example $V_R^{(-)}$.
3. $\mathcal{I}_{P_R}(\tau_W,\tau_R) = 0$ in all other cases.

4.2 Diagonal Topic Transitions

Note that in the unconstrained form presented in Equation 2, we capture transitions between all pairs of topics in the lexicon $\mathbf{T}$. Therefore, given a transition matrix in Figure 3 we need to learn a transition weight for each cell in the matrix.

This approach is clearly infeasible for large open domain lexicons, which can contain millions of unique topics, since there are $\mathbf{T}^2$ possible transitions. Moreover, the transition matrix will be very sparse since there will be relatively few topic pairs with observed transitions. Even for pairs where we do observe transitions, the number of observed examples may be too small to learn a reliable model.

Therefore, we restrict ourselves to learning the transition weights only for the diagonal of the transition matrix (marked in grey in Figure 3). This makes the learning tractable, and leads to a more reliable model with less overfitting, since a diagonal transition is likely to be observed more frequently than a transition between two arbitrary topics.

Also, note that the diagonal transitions mirror the information retrieval approach described in Section 3, where only matches between topics that occur both in the watch and the related video are considered (Equation 1). This facilitates a fair comparison between the performance of these approaches, as discussed in Section 7.

4.3 Loss Miminimization

Given the diagonal topic transition representation described in Section 4.2, the train set $\mathbf{S}_P$ is defined over all the non-discordant pairs of related videos, sampled from user traffic. For the i-th pair $P_i \in \mathbf{S}_P$, a transition vector is defined as

$$\mathbf{x}_i = \{\mathcal{I}_{P_i}(\tau,\tau) \colon \tau \in \mathbf{T}\},$$

such that x_{ij} denotes the j-th diagonal topic transition for the i-th pair of related videos.

Then, we seek a weight vector $\mathbf{w}^*$, which minimizes the l_1- regularized logistic loss over the train set $\mathbf{S}_P$.

Algorithm 1: The *parallel-update* optimization algorithm [12] for learning topic transition weights.

$t = 1$;
repeat
  for *instance* $i = 1, \ldots, |\mathbf{S}_P|$ **do**
    $q^t(i) = \mathcal{L}(\mathbf{w}, \{\mathbf{x}_i\})$;
    for *transition* $j = 1, \ldots, |\mathbf{T}|$ **do**
      $\mu_j^+ = \sum_{i:\ sign(x_{ij})=+1} q^t(i)|x_{ij}|$;
      $\mu_j^- = \sum_{i:\ sign(x_{ij})=-1} q^t(i)|x_{ij}|$;
      $\Delta_j^t = \frac{1}{2}\log\frac{\mu_j^+}{\mu_j^-}$;
      $\mathbf{w}^{t+1} = \mathbf{w}^t + \mathbf{\Delta}^t$;
      $t = t + 1$;
    end
  end
until *convergence* ***or*** *max # of iterations reached*;
return $\mathbf{w}^{t+1}$

Formally, given the loss function over an arbitrary set of examples $\mathbf{S}$, parameterized by a weight vector $\mathbf{w}$

$$\mathcal{L}(\mathbf{w}, \mathbf{S}) = \sum_{\mathbf{x}\in\mathbf{S}} \log(1 + \exp(-\mathbf{w}\cdot\mathbf{x})) + \lambda||\mathbf{w}||_1, \qquad (3)$$

where $\lambda > 0$ is the regularization parameter, we seek an optimal vector $\mathbf{w}^*$ such that

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \mathcal{L}(\mathbf{w}, \mathbf{S}_P). \qquad (4)$$

There is a wide variety of methods that could be employed to minimize the loss function in Equation 3. In this paper, we use the *parallel-update* optimization algorithm, first proposed by Collins et al. [12].

The parallel-update algorithm for finding the optimal diagonal transition weight vector $\mathbf{w}^*$ is outlined in Algorithm 1. The parallel-update algorithm iteratively updates the weight vector $\mathbf{w}^t$ at round t by vector $\mathbf{\Delta}^t$ which is computed using the loss on each individual instance $q^t(i)$.

The main advantage of the parallel-update algorithm is its scalability in the number of examples and the number of features. Note that in Algorithm 1, the $q^t(i)$ and $\mu_j^{\pm}$ parameters are computed independently for each instance i and transition j, which naturally enables parallelization of the weight updates. In addition, the parallel-update algorithm can continuously update the weight vector $\mathbf{w}^*$ as new instances are added to the train set. This property is particularly important in the online setting, where training instances are continuously added based on new user feedback.

4.4 Retrieval with Topic Transitions

Given the optimal diagonal topic transition weight vector $\mathbf{w}^*$ computed by the Algorithm 1, we use it to rank the related video suggestions in response to a watch video V_W. The scoring function is simply given by

$$sc(V_W, V_R) = \sum_{\{w_i\in\mathbf{w}^*\colon\ \tau_i\in V_W\cap V_R\}} w_i \qquad (5)$$

Note that Equation 5 does not require utilizing any additional heuristics such as *idf* weighting, in addition to the transition weights. This is due to the fact the weights $\mathbf{w}^*$ are

optimal for transitioning from watch video with topic τ_i to a related video with topic τ_i, according to the loss criterion defined in Equation 3.

5. SYSTEM OVERVIEW

In this section, we provide a holistic overview of the general architecture of the experimental video suggestion system, which was designed for evaluating the topic retrieval methods described in the previous sections. This system was designed specifically for the purpose of the experiments in this paper, and does not correspond to any system used in production.

We built this experimental system in order to integrate both the co-view based and the topic-based retrieval methods into a single video suggestion system, shown in Figure 4. The purpose of this integration is to demonstrate the benefits of our topic-based video suggestion approach, as compared to the standard co-view based video suggestion approaches [7, 13].

For a given watch video V_W, two retrieval processes are run in parallel. First, we retrieve a set of k related candidates based on the link structure of the co-view graph. See Baluja et al. [7] (surveyed in Section 2.2) for the detailed description of an implementation of such a process. Most generally, this process retrieves all the videos that were most oftenly co-watched with the watch video V_W, regardless of their topicality.

Second, we retrieve top-k ranking candidate videos using a topic-based retrieval process. This process is based either on the weighted topic retrieval or the diagonal topic transitions, described in Section 3 and Section 4, respectively.

This set of $k+k$ highest ranked results from both co-view and topic retrieval processes is sent to the reranking model. For fairness of our experimental setup, we ensure that the number of candidate videos from both co-view and topic retrieval processes are equal.

The final reranking model is out of the scope of this paper. However, for our purposes, it is important to make two general comments about it.

First, the reranking model takes into account a very large number of features based on the watch-related video pair. This makes the reranking model prohibitively expensive to run on the entire video corpus. Instead, the retrieval processes (based on either co-views or topics) provide a small set of tens of potentially relevant video candidates to the reranking model. By augmenting the co-view retrieval with the top results from the topic retrieval, we seek to diversify and improve this candidate set.

Second, to avoid bias in our evaluation, the reranking model excludes any topic-related features, which are used in the topic retrieval process. Therefore, we strive to ensure that the candidates from the topic retrieval process are not given an unfair advantage by the reranking model.

In the experiments in Section 7 we describe how the related video suggestion system design outlined in Figure 4 is used to evaluate the benefits of the topic-based video retrieval using both user simulation and a live traffic experiment.

6. RELATED WORK

In this paper, we propose two retrieval methods for related video suggestion. The first retrieval method, described in Section 3, is based on the standard practices in information retrieval including *idf* term weighting [23] and stopword removal [19] applied to the topical video representations.

The second retrieval method, described in Section 4, is based on learning the optimal transition weights between the watch video and the related video topics. To learn the weights, we use a pairwise classification approach, as proposed by e.g., Burges et al. [9] and Joachims [17], which is common in learning-to-rank applications [18].

The novelty of our approach is due to the fact that we are able to leverage the rich user feedback available in the online setting to learn a large scale model of topic transitions. To this end, we use a parallel-update algorithm, first proposed by Collins et al. [12], which allows continuous and scalable weight learning (see Section 4.3 for details).

Our approach to related video suggestion complements previous work by Baluja et al. [7] that uses co-view information to suggest related videos. As we show in Section 7, our approach can be used to augment the information from a co-view graph, which is especially beneficial for videos with sparse co-view data or rich topical content.

It is important to note the connection between this work and the ongoing research on hybrid recommender systems [4, 10, 15, 31]. Similar to the hybrid recommender systems, the related video suggestion system described in Section 5 combines collaborative information (retrieval from the co-view graph) with content information (topic retrieval) to improve the system performance, especially for the "cold-start" videos with sparse co-view data, or videos with rich topical representations.

Finally our work is also related to some of the current work on supervised topic weighting for video [31] and text [5] retrieval. However, this related work focuses on offline evaluation of the proposed methods. In contrast, we address the challenge of performing evaluation of our methods on live traffic, and demonstrate that our system can be successfully deployed on global scale.

7. EVALUATION

In this section, we empirically evaluate the benefits of the integration of the topic video retrieval methods into the experimental video suggestion system described in Figure 4. To this end, we first describe the methodology and the metrics used in our experiments (Section 7.1). Then, in Section 7.2 we use a simulation method to estimate the effect of the topic-based retrieval on the overall performance of our experimental video suggestion system. Finally, in Section 7.3 we describe the results of a large scale live traffic experiment.

7.1 Experimental Methods

7.1.1 Evaluation Methodology

There are several possible options for evaluating a large scale recommendation system such as the one described in this paper. These options include user simulation based on historical data, user studies and online evaluation [24]. Each of these options has its advantages and limitations, however one important consideration in the choice of the evaluation method is that the test subjects must represent as closely as possible the population of the users in the actual system [24].

This is especially true in the case of large scale web applications that are used by millions of users. In this case,

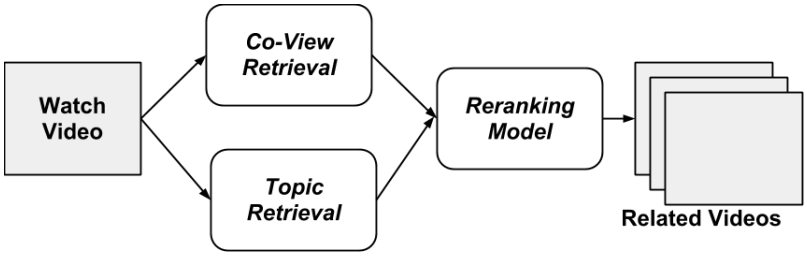


Figure 4: Overview of the related video suggestion system.

obtaining a sample of test subjects that would faithfully represent the real user population is virtually impossible. In addition to the biased sample problem, conducting a user study to evaluate a large scale video recommendation system has several other disadvantages.

First, even given a very detailed set of instructions, it would be difficult for the test subject to judge what would be the best related video to suggest, since this decision is very subjective and may be influenced by factors such as user demographics, geographic location, emotional state and cultural preferences. Even for relatively objective evaluation tasks, such as document retrieval [6] the inter-judge agreement is low. We expect the agreement rate to be even lower for rating video relatedness, which is highly subjective.

Second, as research shows [22], there is often a disconnect between what the subjects really want to watch and what they would like to *have watched*. This leads to a situation where there is little correlation between the explicitly solicited judgments and the observed user behavior in the system.

Therefore, in the next sections we evaluate the performance of the proposed methods using user simulations and a large scale online experiment, and forego evaluation of our methods on manually labeled data.

7.1.2 Metrics

Given the user-centric evaluation method of our system, in this section we address the question of what is the most suitable evaluation metric in this particular setting.

One possible choice of a metric is a click-through rate on the related video suggestions presented to the user by the system. However, research shows that the click-through rate can be highly biased by factors such as position and attractiveness of the presentation [32]. We expect this bias to be very strong in our setting, where the results are presented in ranked order, and each related result is presented as a small snapshot from the video.

Another choice of metric is based on the main functionality of the related video suggestion system, and it measures the watch times of the related videos. Intuitively, a systematic improvement in functionality will generate more relevant suggestions, which will result in a higher user engagement with the system, and lead to longer watch times of the suggested related videos.

Following this intuition, we choose a *watch time* metric, which estimates how much time the user spends watching videos during the session following a click on a related video suggestion. While the watch time metric has its limitations (e.g. it might prefer videos with longer watch times), it is a good proxy for measuring performance, since it faithfully represents the core functionality of the evaluated system.

7.1.3 Retrieval Methods

In this paper we presented two possible ways of integrating topic retrieval into the related video suggestion system. First, in Section 3 we discussed a retrieval algorithm that assigns weights to topics using co-occurence based heuristics. Second, in Section 4 we presented a novel algorithm for directly learning weights on topic transitions.

We evaluate both of these methods by integrating them into the general related video suggestion system architecture as described in Figure 4. First, the highest ranked results produced by one of the two proposed retrieval methods are introduced into the reranking model. Then we measure the changes in the overall system performance, using either a simulation experiment (Section 7.2) or an experiment using live traffic (Sectrion 7.3).

In the next sections, we refer to the retrieval algorithm presented in Section 3 as `IRTopics`, since it makes use of information retrieval heuristics. We refer to the retrieval algorithm from Section 4 as `TransTopics`, since it is based on learning transitions between the topics.

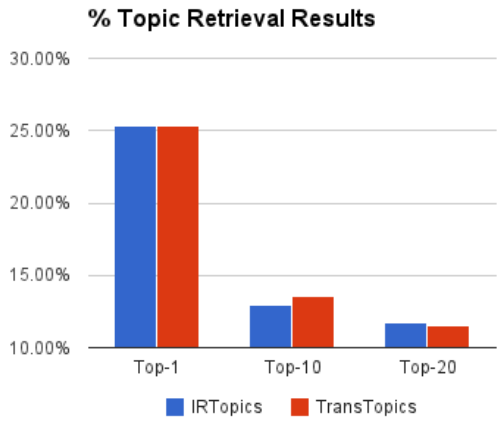
7.2 User Simulation

In this section we describe a user simulation method for estimating the performance of our retrieval methods. We exploit the reranking model described in Section 5 to simulate user interaction with the system.

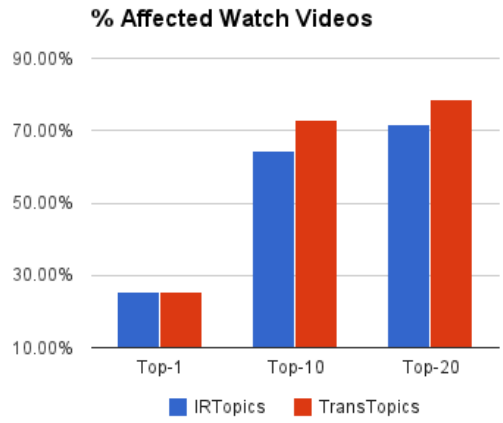
Since the reranking model is trained to optimize the system performance (in terms of click-through rate and watch time) on live traffic, we use it to simulate a behavior of a typical user in the system. Then we measure how many of the results returned by the topic retrieval method will be added by the simulated user to the top related results, compared to a system that only uses co-view retrieval. By system design, if no results from the topic retrieval are selected by the simulated user, there is no benefit from performing this retrieval, since none of the results will be shown to the real users.

There are two things we are interested in measuring. First, we measure how many new results our method introduces that were not previously returned by the co-view retrieval approach. Second, and more importantly, we want to observe how many of these results are actually considered as relevant related videos by the simulated user (i.e., positioned at high ranks by the reranking model).

We run the user simulation for a large sample of videos. Figure 5(a) shows the percentage of new videos ranked among the top-K results by the simulated user that were intro-



(a) Percentage of new related videos.



(b) Percentage of watch videos with new related videos.

Figure 5: User simulation results.

duced by the topic retrieval for either the `IRTopics` or the `TransTopics` methods. Figure 5(b) shows the percentage of watch videos that have at least one new related video among the top-K results as a result of topic retrieval.

As Figure 5(a) shows, there is a significant number of results added at the top ranks as a result of topic retrieval. Figure 5(b) demonstrates that these results are spread among watch videos. More than 70% of watch videos are affected by our retrieval, and have at least one new top-20 result coming from the topic retrieval stage.

Both of the proposed methods retrieve roughly the same number of new results, with the `TransTopics` method introducing slightly more results at the higher ranks (13.6% compared to 12.9% at the top ten results). This potentially indicates the higher relevance of the new results introduced by the `TransTopics` method.

Figure 5(b) shows a similar trend. At top ten results, the `TransTopics` method affects 73.1% of watch videos, compared to the 64.6% affected by the `IRTopics` method.

While the simulation method described in this section is suitable for the purposes of validation and testing different variants of the proposed methods, it does not provide a definitive answer whether the proposed topic-based videos will indeed have a positive effect on the actual user experience. To this end, we conduct a large scale live experiment that is described in the next section.

7.3 Live Experiment

7.3.1 *Live Experiment Summary*

To evaluate the performance of the `IRTopics` and the `TransTopics` methods, we conducted a large scale experiment on a random sample of live YouTube traffic. The experiment was run during a single month in 2013, and affected related video suggestions returned for millions of watch videos. The topic weights for both retrieval methods were updated on the daily basis during the experiment, according to the method descriptions in Section 3 and Section 4, respectively.

Figure 6 presents a summary of the live experiment findings. The differences in Figure 6 are reported with respect to a baseline system that does not employ a topic retrieval stage (but does employ the co-view retrieval and the reranking model, as described in Figure 4).

In Figure 6 we report the differences in three metrics related to watch time. The first metric is the *watch time* itself (as described in Section 7.1). The second metric is the *completion rate*, which measures how many of the suggested videos were fully watched from start to finish. The third metric is the *abandonment rate*, which measures the fraction of watch videos for which no related videos were watched.

As can be seen in Figure 6, the addition of the topic retrieval stage to the related video suggestion system results in improvements in all three watch metrics: watch time and completion rate increase, while the abandonment rate decreases. As the confidence intervals shown by the error bars in Figure 6 demonstrate, these improvements are statistically significant.

In absolute metrics, the `TransTopics` method achieves overall 1.3% increase in watch time over the baseline setup that does not employ topic retrieval. This is an impressive increase, given billions of hours of video watched monthly on YouTube [2].

In addition, the `TransTopics` method is significantly more effecitve compared to the `IRTopics` method. % change in the watch time is 80% higher for the `TransTopics` compared to the `IRTopics`. Similarly, % change in the completion rate is more than double, and the % change in the abandoment rate drops by more than 90% when comparing the `TransTopics` method to the `IRTopics` method.

These effectiveness improvements highlight the importance of directly learning topic transitions from user feedback. As Figure 6 clearly demonstrates, the learned weights in the `TransTopics` method can significantly improve the system performance compared to the hand-crafted heuristic weighting in the `IRTopics` method.

7.3.2 *Breakdown by Video Type*

In addition to the summary presented in the previous section, it is interesting to further analyze the performance of our methods by video type. In Table 1 we break down the changes in the watch time metric by *video category* (specified by the uploader of the video) and *video age*.

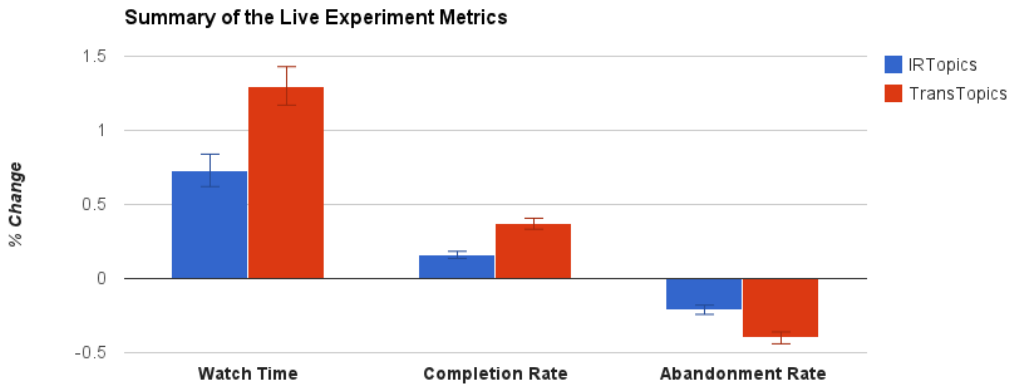


Figure 6: Summary of the live traffic experiment metrics: *watch time*, *completion ratio* and *abandonment rate*. 95% observed confidence intervals are shown by the error bars.

	IRTopics	TransTopics
Video Category		
Music	-0.64% ($\pm 0.09\%$)	$+0.28\%$ ($\pm 0.09\%$)
Gaming	$+0.86\%$ ($\pm 0.68\%$)	$+1.14\%$ ($\pm 0.66\%$)
News	$+1.61\%$ ($\pm 0.41\%$)	$+3.53\%$ ($\pm 0.41\%$)
Science and Technology	$+2.43\%$ ($\pm 0.5\%$)	$+3.79\%$ ($\pm 0.51\%$)
Pets and Animals	$+3.70\%$ ($\pm 0.68\%$)	$+4.16\%$ ($\pm 0.66\%$)
Video Age		
$<$ 1 month	$+0.99\%$ ($\pm 0.26\%$)	$+3.34\%$ ($\pm 0.26\%$)
1 month – 1 year	$+0.50\%$ ($\pm 0.11\%$)	$+2.24\%$ ($\pm 0.11\%$)
$>$ 1 year	$+0.87\%$ ($\pm 0.07\%$)	$+1.06\%$ ($\pm 0.08\%$)

Table 1: Live traffic experiment watch time metric breakdown by video category and age. 95% observed confidence intervals are shown in the parentheses.

As our retrieval methods are topic based, we expect them to be most beneficial for videos with rich topical content and videos with low co-view signal. The results in Table 1 confirm this hypothesis.

For categories with richer topical representations like *News* and *Science and Technology*, the improvements obtained by our method are approximately three times higher than the average improvement in Figure 6. The highest improvements are obtained for the *Pets and Animals* category, which has many tail videos with little co-view information. On the other hand, for *Music* and *Gaming* categories, which have more popular videos with co-view data, the topic retrieval has smaller positive benefits (or even a slight negative impact for the `IRTopics` method in the *Music* category).

Similarly, for fresh videos (e.g., less than a month old), our improvements are more significant (especially for the `TransTopics` method) than for the older videos. This is due to the fact that the fresher videos have a weaker co-view signal.

The results in Table 1 demonstrate that our topic retrieval methods improve the related video suggestions by augmenting the co-view results with fresher and more topically relevant videos. In agreement with the general results in Figure 6, the `TransTopics` method is always significantly more effective than the `IRTopics` method across different video categories and age groups.

8. CONCLUSIONS

In this paper, we focused on the task of related video suggestion, commonly found in many online applications. The majority of the current approaches to related video suggestion are based on the collaborative filtering analysis. Most generally, the collaborative filtering approaches suggest videos that are likely to be co-viewed with the currently watched video.

While such suggestions work well for popular videos with many views, it is less applicable in situations where little or no view data is available. This includes fresh videos, or tail videos with few views. To address this challenge, we propose a hybrid approach to video suggestion, which combines the video content with the co-view information to improve the related video suggestions.

To this end, we represent the video using a set of topics mined from various sources including the video metadata, frequent uploader keywords, common search queries, playlist names and Freebase entities. In order to achieve an effective topical video representation, we assign weights to each topic, and propose two approaches for topic weighting.

The first approach is based on the standard information retrieval heuristics such as topic frequency and inverse document frequency weighting and stopword removal. The downside of this approach is that it does not directly take into the account the user behavior.

Accordingly, we develop a second approach, which leverages the implicit user feedback (such as video co-watches) available in the online setting. This feedback is used for supervised learning of the optimal topic weights. Unlike the standard collaborative filtering analysis, our approach takes into account topic-to-topic rather than video-to-video co-view information. This enables suggesting a related video even if it was never explicitly viewed with the watched video.

We empirically evaluate these two topical representation methods in a large scale live experiment on YouTube traffic. We find that both of the representations have a significantly positive effect on the quality of video suggestions, compared to the standard collaborative filtering approach. This effect is especially visible for videos with rich topical representation such as videos in the *News* and *Science and Technology* categories (more than 3.5% increase in watch time), and for fresh videos with sparse co-view data (more than 3% increase in watch time).

Furthermore, our evaluation demonstrates that learning topic weights from user feedback can increase the user engagement (as compared to the collaborative filtering baseline) by more than 80% over the standard information retrieval representation. This demonstrates the importance of incorporating user feedback in content representation.

While this paper focuses on the task of related video suggestion, the findings of this work are broad, and are applicable in a variety of large scale retrieval and recommendation systems that employ topical content representations. As this work shows, learning topic weights from user feedback have the potential to significantly improve performance over the standard non-supervised weighting approaches.

9. ACKNOWLEDGMENTS

The authors would like to thank Ajith Ramanathan, Jianming He, Su-Lin Wu, Tomas Lloret Llinares and Tamas Sarlos for their advice and assistance in conducting the experiments described in this paper.

10. REFERENCES

[1] Give YouTube topics on search a whirl. `http://youtube-global.blogspot.com/2010/11/give-youtube-topics-on-search-whirl.html`.

[2] Youtube – statistics. `http://youtube.com/yt/press/statistics.html`.

[3] Youtube data API - searching with Freebase topics. `https://developers.google.com/youtube/v3/guides/searching_by_topic`.

[4] A. Ahmed, B. Kanagal, S. Pandey, V. Josifovski, L. G. Pueyo, and J. Yuan. Latent factor models with additive and hierarchically-smoothed user preferences. In *Proceedings of WSDM*, pages 385–394, 2013.

[5] B. Bai, J. Weston, D. Grangier, R. Collobert, K. Sadamasa, Y. Qi, O. Chapelle, and K. Weinberger. Supervised semantic indexing. In *Proceedings of CIKM 2009*, pages 187–196, 2009.

[6] P. Bailey, N. Craswell, I. Soboroff, P. Thomas, A. P. de Vries, and E. Yilmaz. Relevance assessment: are judges exchangeable and does it matter. In *Proceedings of SIGIR*, pages 667–674, 2008.

[7] S. Baluja, R. Seth, D. Sivakumar, Y. Jing, J. Yagnik, S. Kumar, D. Ravichandran, and M. Aly. Video suggestion and discovery for youtube: taking random walks through the view graph. In *Proceedings of WWW*, pages 895–904, 2008.

[8] A. Z. Broder, D. Carmel, M. Herscovici, A. Soffer, and J. Zien. Efficient query evaluation using a two-level retrieval process. In *Proceedings of CIKM*, pages 426–434. ACM, 2003.

[9] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *Proceedings of ICML*, pages 89–96, 2005.

[10] R. Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, Nov. 2002.

[11] B. Chen, J. Wang, Q. Huang, and T. Mei. Personalized video recommendation through tripartite graph propagation. In *Proceedings of MM*, pages 1133–1136, 2012.

[12] M. Collins, R. E. Schapire, and Y. Singer. Logistic regression, adaboost and bregman distances. *Machine Learning*, 48(1-3):253–285, Sept. 2002.

[13] J. Davidson, B. Liebald, J. Liu, P. Nandy, T. Van Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, B. Livingston, and D. Sampath. The youtube video recommendation system. In *Proceedings of RecSys*, RecSys '10, pages 293–296, New York, NY, USA, 2010. ACM.

[14] M. Fontoura, V. Josifovski, J. Liu, S. Venkatesan, X. Zhu, and J. Zien. Evaluation strategies for top-k queries over memory-resident inverted indexes. *Proceedings of the VLDB Endowment*, 4(12):1213–1224, 2011.

[15] A. Gunawardana and C. Meek. A unified approach to building hybrid recommender systems. In *Proceedings of RecSys*, pages 117–124, 2009.

[16] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of SIGIR*, pages 119–126, 2003.

[17] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of KDD*, pages 133–142, 2002.

[18] H. Li. Learning to rank for information retrieval and natural language processing. *Synthesis Lectures on Human Language Technologies*, 4(1):1–113, 2011.

[19] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[20] P. Over, G. Awad, J. Fiscus, B. Antonishek, M. Michel, A. F. Smeaton, W. Kraaij, G. Quénot, et al. TRECVID 2012 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *TRECVID 2012-TREC Video Retrieval Evaluation Online*, 2012.

[21] F. Radlinski, M. Kurup, and T. Joachims. How does clickthrough data reflect retrieval quality? In *Proceedings of CIKM*, pages 43–52, 2008.

[22] D. Read, G. Loewenstein, and S. Kalyanaraman. Mixing virtue and vice: Combining the immediacy effect and the diversification heuristic. *Journal of Behavioral Decision Making*, 12(4):257–273, 1999.

[23] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

[24] G. Shani and A. Gunawardana. Evaluating recommendation systems. In *Recommender systems handbook*, pages 257–297. Springer, 2011.

[25] V. Simonet. Classifying youtube channels: a practical system. In *Proceedings of WOLE 2013, in Proceedings of WWWW companion*, pages 1295–1304, 2013.

[26] A. Singhal, C. Buckley, and M. Mitra. Pivoted document length normalization. In *Proceedings of SIGIR*, pages 21–29, 1996.

[27] C. G. M. Snoek and M. Worring. Multimodal video indexing: A review of the state-of-the-art. *Multimedia Tools and Applications*, 25(1):5–35, Jan. 2005.

[28] T. Tsikrika, C. Diou, A. P. de Vries, and A. Delopoulos. Image annotation using clickthrough data. In *Proceedings of CIVR*, pages 14:1–14:8, 2009.

[29] C. Vondrick, D. Patterson, and D. Ramanan. Efficiently scaling up crowdsourced video annotation. *International Journal of Computer Vision*, pages 1–21. 10.1007/s11263-012-0564-1.

[30] J. Weston, S. Bengio, and N. Usunier. Large scale image annotation: learning to rank with joint word-image embeddings. *Machine Learning*, 81(1):21–35, Oct. 2010.

[31] B. Yang, T. Mei, X.-S. Hua, L. Yang, S.-Q. Yang, and M. Li. Online video recommendation based on multimodal fusion and relevance feedback. In *Proceedings of CIVR 2007*, CIVR '07, pages 73–80, 2007.

[32] Y. Yue, R. Patel, and H. Roehrig. Beyond position bias: examining result attractiveness as a source of presentation bias in clickthrough data. In *Proceedings of WWW*, pages 1011–1018, 2010.