Analysis of User-generated Content for Improving YouTube Video Recommendation

Michele Galli, Davide Feltoni Gurini, Fabio Gasparetti[*] , Alessandro Micarelli, Giuseppe Sansonetti
Roma Tre University
Via della Vasca Navale 79 - Rome, 00146 Italy
{michele.galli,feltoni,gaspare,micarel,gsansone}@dia.uniroma3.it

ABSTRACT

Everyday video-sharing websites such as YouTube collect large amounts of new multimedia resources. Comments left by viewers often provide valuable information to describe sentiments, opinions and tastes of users. For this reason, we propose a novel re-ranking approach that takes into consideration that information in order to provide better recommendations of related videos. Early experiments indicate an improvement in the recommendation performance.

Categories and Subject Descriptors

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

Keywords

Recommender systems, Web 2.0, YouTube

1. INTRODUCTION

YouTube is the world's most popular web video community used by 1 billions unique users world wide each month[1]. Four billions of videos are viewed per day, with 100 hours of new ones uploaded every minute. Sifting through this large repository of multimedia resources poses unique challenges for the user.

The YouTube user interface provides, given the current video l_{id}, a list of recommendations as shown in Fig. 1. YouTube selects those recommendations based on an algorithm that considers signals from a variety of sources including the user's favorite, watched and liked videos [4]. These signals are combined for ranking the list of *related* videos compiled by monitoring what other people usually watch next. By exploring this related-video graph, a candidate list is built. Characteristics about the videos (e.g., views and ratings) and the similarities of the videos with the history of videos watched by the user are combined to rank the candidate resources. A trade-off between relevance and diversity across categories builds up the related video list $L_{id} = (l_1, l_2, \ldots, l_n)$. As a result, the user-generated comments that are shown below the video are not taken into consideration. Although these user interactions are often short and noisy, they have the chance to represent valuable information about user interests, tastes and, more in general, debate topics about the videos.

Figure 1: The YouTube website with metadata and recommended videos highlighted.

Related video lists can host a large number of suggestions, i.e., up to 40. Our hypothesis is that two videos may be related if they give rise to similar reactions and sentiments from viewers. This sort of implicit relationship between multimedia resources might improve the original YouTube ranking in a way that better matches the user expectations. In this paper we propose a re-ranking method that, for each video, generates a new ordered list of videos proposed by the YouTube traditional recommender.

2. THE PROPOSED VIDEO RECOMMENDATION

Given the l_{id} video, the YouTube Data API[2] allows us to retrieve up to 1000 comments $C_{l_{id}} = \{c_1, c_2, \cdots\}$.The API provides us also the top 25 related videos. We filter too short

[*]Contact author email: gaspare@dia.uniroma3.it
[1]http://www.youtube.com/yt/press/statistics.html (Accessed: 2 July 2015)
[2]https://developers.google.com/youtube/ (Accessed: 2 July 2015)

RecSys 2015 Poster Proceedings, September 16-20, 2015, Vienna, Austria.
.

comments and the ones with obscene or profane language. A Bayesian classifier trained on a subset of spam comments help us to filter out the less relevant content.

A keyword-based approach [2] identifies the words that express a sentiment, assigning them a score in $[0, 1]$ to each of the following dimensions: *positivity*, *negativity*, and *objectivity*. In particular, given a comment $c_i \in C_{l_{id}}$ we sum up all the positivity scores and then subtract the negativity ones. The obtained normalized real value is encoded in a categorical feature by linearly discretizing it to 5 intervals so that each comment is assigned to one of the following classes: very positive, positive, neutral, negative, very negative. Those classes are also the five dimensions of a vector space model, where the sentiment vector:

$$\overrightarrow{v_{l_{id}}^{(ss)}} = (v_{1,id}, v_{2,id}, v_{3,id}, v_{4,id}, v_{5,id}) \quad (1)$$

is calculated by summing up the occurrences of the very positive classes for the dimension $v_{1,id}$, positive occurrences for $v_{2,id}$, neutral occurrences for $v_{3,id}$, and so forth. The same procedure is followed for each video $l_j \in L_{id}$ by analyzing the set of comments associated with l_j. We obtain n vectors $\overrightarrow{v_{l_j}^{(ss)}}$ that can be compared by means of a cosine similarity measure with $\overrightarrow{v_{l_{id}}^{(ss)}}$. The related video l_j will thus have a sentiment-based similarity $r_{id,j}^{(ss)} \in [0, 1]$.

A second step extracts named entities (e.g., persons, locations) and nouns from each comment by means of the Stanford Named-entity recognizer and Part-of-Speech tagger, respectively. As with the previous procedure, two vectors, $\overrightarrow{v_{l_j}^{(ne)}}$ and $\overrightarrow{v_{l_j}^{(pos)}}$, are obtained for each video l_j in $L_{id}^{(y)}$ by summing up the contribution of the different comments. The two vectors $\overrightarrow{v_{l_{id}}^{(ne)}}$ and $\overrightarrow{v_{l_{id}}^{(pos)}}$ are also computed. The dimensions of the vectors are distinct named entities and nouns that appear in the analyzed user-generated data. A cosine similarity measure assigns the scores $r_{id,j}^{(ne)}$ and $r_{id,j}^{(pos)}$ between l_{id} and l_j videos, respectively, for the named entity and noun comparisons.

The last step calculates the final rank for the video j by linearly combining the three measures:

$$r_{id,j} = \alpha_1 r_{id,j}^{(ss)} + \alpha_2 r_{id,j}^{(ne)} + \alpha_3 r_{id,j}^{(pos)} \quad (2)$$

where the three α values are set to the α_0 constant.

3. EVALUATION

A total of 8 persons were involved, mostly students of CS courses, all usual users of the YouTube service. A Java application has been developed to assist them during the evaluation. We asked them to select 10 videos $V = \{v_1, \ldots, v_{10}\}$ from their watched history, the recommendations on the YouTube homepage or the subscribed channels. For each video $v_i \in V$ the application obtains its related YouTube videos L_{v_i}. A new ordered list L'_{v_i} is built by downloading the comments and running the proposed approach on them. A randomized list is proposed to each user that was asked to evaluate her interests in watching each single video with a five-level Likert scale. The Normalized discounted cumulative gain (nDCG) is evaluated both for the YouTube list L_{v_i} and the new ranked one L'_{v_i}. After computing the measure for each video we averaged them to obtain an overall performance evaluation. The YouTube recommender obtains a nDCG of 0.829 while the proposed approach reaches 0.858 with an improvement of 3.51% (p-value$<$0.05).

4. RELATED WORKS

To the best of our knowledge, our work makes the first attempt to analyze user comments in the video recommendation domain. Shmueli *et al.* [6] analyze users' co-commenting patterns for predicting, for a given user, suitable news stories that she likely comment on. A similar approach is focused on the news recommendation by Messenger and Whittle [5]. Sergiu *et al.* [3] explore the effectiveness of comments and other social signals for the video retrieval task, that is, when a user query must be elaborated.

5. CONCLUSIONS AND FUTURE WORK

Whereas the obtained benefits in the re-rank of YouTube related videos is limited, the statistical significance of findings let us think that a textual comment mining approach should be considered for future investigations. Much of the computation can be implemented offline, while the basic cosine similarity calculus has limited complexity.

More experiments are undergoing to better understand the relationship between the kinds of opinions and sentiments expressed by the users and the categories of the videos. By collecting a large training dataset, it is possible to dynamically assign different weights to the three parameters of Eq. 2. Temporal dimension is a further element to consider [1]. There are many videos for which YouTube is not able to compute a reliable set of related videos due to the scarcity of user activities. It is interesting to understand if the proposed approach can be successfully implemented even for new videos that have collected a right number of comments, partially addressing the data-sparsity issue due to the scarcity of user activity records.

6. REFERENCES

[1] G. Arru, D. Feltoni Gurini, F. Gasparetti, A. Micarelli, and G. Sansonetti. Signal-based user recommendation on twitter. In *Proc. of WWW '13*, pages 941–944, Republic and Canton of Geneva, Switzerland, 2013.

[2] S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In N. Calzolari and et al., editors, *LREC*. European Language Resources Association, 2010.

[3] S. Chelaru, C. Orellana-Rodriguez, and I. Altingovde. How useful is social feedback for learning to rank youtube videos? *World Wide Web*, 17(5):997–1025, 2014.

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

[5] A. Messenger and J. Whittle. Recommendations based on user-generated comments in social media. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE SocialCom*, pages 505–508, Oct 2011.

[6] E. Shmueli, A. Kagian, Y. Koren, and R. Lempel. Care to comment?: Recommendations for commenting on news stories. In *Proc. of WWW '12*, pages 429–438, New York, NY, USA, 2012. ACM.