Relevance-Based Ranking of Video Comments on YouTube

Conference Paper · May 2013

DOI: 10.1109/CSCS.2013.87

2 authors:

Andrei Serbanoiu

1 PUBLICATION **7** CITATIONS

SEE PROFILE

Traian Rebedea

Polytechnic University of Bucharest

121 PUBLICATIONS **610** CITATIONS

SEE PROFILE

Relevance-Based Ranking of Video Comments on YouTube

Andrei Serbanoiu, Traian Rebedea
Faculty of Automatic Control and Computers
University Politehnica of Bucharest
Bucharest, Romania
serbanoiu.andrei@gmail.com, traian.rebedea@cs.pub.ro

***Abstract*—There are many web platforms that are used to share non-textual content, such as videos, images, animations, etc. and that allow users to add comments for each item. YouTube is probably the most popular of them, with billions of videos uploaded by its users, but also with billions of comments for all these videos. While most of the videos only have a couple of comments, the most debated ones have tens of million comments. However, the platform does not provide any mechanism to filter the comments in order to get the most relevant ones, as the only provided ordering is in descending order of the publishing date. We propose a ranking mechanism for these comments that tries to determine the relevance of each comment to the individual video by automatically linking it to relevant web pages with textual information. Three different ranking methods are discussed and their results are presented in order to offer a comparison among them.**

Keywords—Relevance; Ranking; Text Classification; Latent Dirichlet Allocation; Video Comments

I. INTRODUCTION

Text categorization is one of the most popular tasks in machine learning and natural language processing and it consists in automatically labeling "a set of documents into categories from a pre-defined set" [1]. It has been used in a wide number of tasks, such as spam identification, classification of web pages, authorship attribution, essay assessment, automatic language recognition, etc. More recently, text categorization has started to be used with social data and informal conversations as well, for example in order to identify social spammers of various kinds [2], to classify users part of online political debates [3] or even to identify sexual predators in online chats [4]. However, in these situations other data has to be combined with the text features in order to find a good classification method. This may be user-related information or even interaction-related data between various users.

This paper focuses on classifying and ranking a special class of texts: comments on web platforms that contain non-textual items such as video, audio files or images. The specific application was developed for the YouTube (http://www.youtube.com) platform, which is the largest such online platform both in number of uploaded materials and in the number of users.

The main characteristics of the comments and the aspect that makes this task harder than other text categorization problems is that, unlike documents that have thousands of words from which to extract relevant information, comments only have a very small number of words – sometimes less than 10, on average of the order of tens. These features are used to take the decision when classifying the comments as relevant or not and then for ranking the relevant comments and presenting them to the users in a specific order provided by the rank.

In the following sections we will present a method for overcoming these problems as we attempt to classify and then rank comments according to a measure of relevance for the video they are addressed to. In this context, relevance is evaluated with respect to the information collected from other online sources about the video commented upon.

The paper continues as follows. Section 2 presents related work to classification and ranking of YouTube comments. We continue with a broader presentation of the problem we propose to solve and introduce a possible solution in section 3. The details of the pre-processing stage are presented in section 4, while the next section shall present three distinct methods, but that share similar features, for computing the relevance of a text comment. Section 6 contains a comparison of the results obtained with the aforementioned methods, while the last section is reserved for concluding remarks.

II. RELATED WORK

Although YouTube has been used as a source of previous research and there have also been attempts to use analyze the comments posted for videos, to our knowledge there is no work that goes exactly in the direction proposed in this paper. However, we shall present the most similar problems and their solutions.

For example, some studies have conducted an in depth analysis of YouTube comments in order to learn more about the community feedback upon comments, and also to find the dependency between high community comment ratings and language or sentiment orientation [5]. They analyzed a set of about 6 million YouTube comments and ratings. By using SentiWordNet [6] they attached a sentiment score to every comment and finally arrived at the conclusion that community feedback along with term features in comments can be used to

automatically determining the community acceptance of comments. Their research was not confined only to music videos and offers a broader perspective on the subject.

One of the interesting thing that were determined is that the features such as top terms, terms most commonly associated with high-rated comments and the terms for unaccepted comments, in combination with the sentiment analysis module, can be used to predict community acceptance. This led us to believe that a preliminary filtering phase based on patterns or common words found in low-rated comments, used in conjunction with comparing the comment text with a reference corpus for relevance, in their case with the tags associated with the video, would yield good results for the problem of finding the most relevant comments in the corpus as well.

Moreover, as YouTube comments and other types of social media content are very noisy, an important attention must be paid to building a robust pre-processing stage [7]. They make use of a comment-term matrix to train a supervised classification model and, by using a previously annotated corpus of both clean and noisy comments, along with a bag of words model, the authors compute a score for each comment. This labeled data, along with the relevance score computed beforehand are used to train a supervised classification model that will learn the underlying classification rules to predict the binary relevance (they use a binary classifier to state if comment is clean or noisy) of each new comment retrieved from YouTube.

This work was useful, as it used a two-stage classification with a Naïve Bayes classifier combined with Decision Tree one that might also be useful for our task as well. Furthermore, we observed that approaches that are used for spam detection yield very good results in such an environment as well. This led us to believe that by using a classifier such as a neural network would provide good results in the pre-processing stage in order to eliminate spam comments which are clearly not relevant for any video. However, we avoided using a classifier in the final stage, that of raking the comments according to their relevance, because we found no measure of accuracy that could be used. We also wanted to include external references for the relevance scoring process as opposed to just using intrinsic properties of the comments.

Other results, which inspired the pre-processing stage, have been adapted from a solution to the problem of comment ranking in the social context [8]. The authors propose an interesting set of features that might be used when classifying comments as spam or legitimate. They use a number of features such as comment visibility, the user reputation as well as some interesting content-based features. The content features are: comment length, comment complexity – as a measure of the entropy of the words in the comment, the number of upper case words as well as the "comment informativeness". The later attempts to capture the uniqueness of the comment relative to other comments associated to the same social web object. This is done by using a variation of the standard tf-idf approach.

III. THE PROBLEM AND PROPOSED SOLUTION

YouTube is one the most visited web sites in the world and it attracts millions of comments per week from its users. All these comments should be related to the information presented in the video that it is addressed to. However, as can be easily be noticed from Fig. 1, many of these comments are purely social, contain advertising, bad language and are not related to the content of the video. Therefore, they are not useful for many people that may be interested to find interesting opinions about the watched video by reading the comments.

Fig. 1. Sample of comments presented for a video on YouTube

After assessing the problem and after evaluating the state of the art in this domain, we arrived to the conclusion that the best approach to arrive at an optimal solution would be to use a system with the architecture presented in Fig. 2. The application represents a mix of open-source libraries and external resources, aggregated into a system designed to produce a good comment relevance classifier. The main characteristic of the system is the division of the processing into two main stages:

- A *preprocessing stage* for removing spam and other types of comments that are considered irrelevant for any video.
- A *relevance ranking stage* (called *relevance classifier* in the figure) that assigns a score to each comment that was not discarded by the first stage.

The system uses the YouTube service to collect comments that will be later classified in relation with reference articles obtained from the Wikipedia, Allmusic and Lyrics resources. As shown in Fig. 2, the retrieved comments are not fed directly to the relevance classifier, but rather get pre-processed by the Weka module. This module represents a collection of classifiers, trained using the open-source solutions available in the data mining solution Weka (http://www.cs.waikato.ac.nz/ml/weka/). For this stage, we have chosen a neural network classifier in order to separate items that are considered to have a very low relevance.

After preprocessing takes place, language detection is performed as the aim of this work is to correctly classify comments written in the English. After that, the Mallet module built upon the Mallet library (http://mallet.cs.umass.edu/) is used to apply LDA (Latent Dirichlet Allocation) as a topic detection algorithm, in order to extract relevant topics from

various pieces of text as explained in the next section. These topics are then enriched by the WordNet module. This last module adds semantic related terms to enrich the list of selected topics.

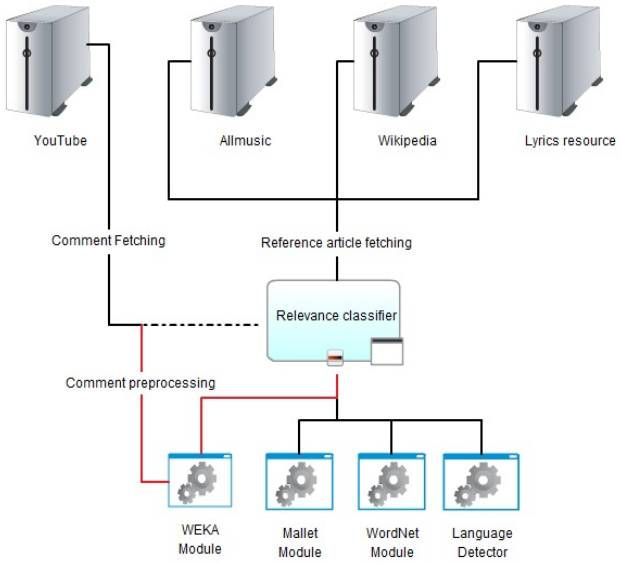

Fig. 2. The architecture of the comment ranking system

Finally, the relevance classifier (or ranker) uses the data provided by all the other modules and, by applying a weighing sum function over a number of features, it provides the final relevance score for the given textual items. In the following sections we will present in greater details the way the architecture evolved before reaching the final form presented above, the inner workings of all the modules used, the way all the modules couple as well as samples of algorithm results that will prove the implementation direction selected is the right one.

The method that produces the ranking of the comments can be summed up as follows:

```
comments = fetchYouTubeComments();
comments = filterComments(comments);
commentTopics = createCommentTopics(comments);
resources  = getResources(wikipedia,allmusic,lyrics);
for(int i=0;i<commentTopics.length;i++)
{
        computeRelevance(commentTopics[i], resources);
}
```

Although it may seem simple and straight forward from this high level perspective, the algorithm poses interesting problems related to the filtering phase and to the resource processing step. Choosing the right features for the filtering classifier, finding the appropriate and trustworthy resources to use as references for relevance are not trivial tasks. But by studying the phenomena, as well as learning from previous work, the proposed solution is a good choice that offers interesting results that should be useful to the final user.

IV. Comment Pre-processing Stage

YouTube video comments were obtained by integrating the YouTube Data API which is publicly available. For the purpose of this work, only the first 100 comments for each video were used, although the API provided means of retrieving up to 1000 of them. This limitation was imposed because the topic extraction process is tedious and it takes a long time to run the algorithm. Nevertheless the results are still of value and prove that this approach yields useful comments to be presented to the users.

The second step after obtaining the comments was to filter them, removing all of the ones written in a language different from English. This was also done using a free Java library JLangDetect (http://www.jroller.com/melix/tags/jlangdetect) that proved to have an accuracy that is satisfying for the task at hand. It should be noted that finding the correct language for small pieces of text, such as comments, always yields poorer results than for longer documents.

After this step, we extracted the main topics from each comment using the Mallet library – for topic extraction – and Wordnet – for enriching the topics with similar concepts. The Mallet library provided a maximum of 5 topics per comment and, because we considered that the number of topics alone would not be enough for the evaluation of the relevance, some other words were added to the topics list. The words were obtained from Wordnet and represent synonyms and hypernyms of the topics extracted. This created a better set of features for each comment, enabling us to assess better the relevance of the comments.

A. YouTube Data API

The YouTube Data API (https://developers.google.com /youtube/) allows programs to perform many of the operations available on the YouTube website. They provide the user with the possibility of searching for videos, retrieving standard feeds, and seeing related content. A program can also authenticate as a user to upload videos, modify user playlists, and more. The YouTube Data API is intended primarily for developers who are used to programming in server-side languages and it is useful for sites or applications that wish to have a deeper integration with YouTube. It also gives programmatic access to the video and user information stored on YouTube. The API also comes with limitations regarding the number of queries per time interval an IP can make using an authentication key. In our solutions, it has been used to extract a list of comments from each of the analyzed videos.

B. Pre-classification of Comments

After initial tests, we observed that a very large part of the execution time was spent extracting features from comments, and this would pose problems if we chose to run the algorithm on a larger set of input data. To overcome this problem we created a classification system based on a neural network that, by using a set of simple linguistic features, would make the feature extraction stage less tedious by reducing the number of comments this stage will be applied upon.

The proposed classifier is a binary Multilayered Perceptron provided by the Weka library which is a collection of machine

learning and data mining algorithms which are exposed for developers as a Java library. The following features have been used for classifying comments into spam (or irrelevant) and relevant regardless of the video:

- *Number of non-ASCII characters*: irrelevant comments tend to have more non-ASCII characters then relevant ones.
- *Number of capital letters*: irrelevant comments have a higher capital letter to words ratio then relevant ones.
- *Number of new-lines*: comments spread on too many lines tend to be irrelevant.
- *Number of digits*: it has been observed that there is a certain class of irrelevant comments that have a high digit number related to the comment length.
- *Number of trivialities*: based on a list of swear-words and by computing the Levenshtein distance, trivialities are a good indicator of irrelevant content.
- *Number of words in comment*: the word count provides a helpful indicator of relevance, as a content-full message can only be expressed by using a large enough number of words.
- *Mean word length*: it is known that higher character/word count can indicate a more complex dialogue and thus can be a good indicator of relevance.
- *The number of punctuation marks*: a coherent message has a rather low word to punctuation mark ratio.
- *Common spam text count*: based on a list of frequent spam text patterns this is a strong indicator of comment spam.

The classifier was then trained on a corpus of 200 comments chosen to contain a wide spread of both relevant and irrelevant/spam comments. Examples of relevant comments contained in the training set are:

- "I love this video because? it reminds me of how my ex girlfriend wants me back and apologizes for cheating on me but this song has as message that sometimes it really is to late to apologize. :)",
- "he didn't write the lyrics, ryan tedder of onerepublic did. the song was already written pretty much exactly like this then timbaland remixed it so it was a bit more up tempo and added the eh eh eh bits...?",
- "great! Asolutely great! You know this kind of songs you hear once and never forget I can listen to this one for over 100 times in a row. She looks absolutely gorgeous with her straight hair. Does anyone know one of her songs which is as good as this one?"

A subset of the irrelevant comments contained in the training set are presented next:

- "IF YOU LIKE DIRTY DIANA SONG THE SINGER " STEFANO GIORGINI " DID A GREAT? REMAKE STEFANO IS A VERY GOOD SINGER SONGWRITER I THINK YOU WILL LIKE HIS VERSION JUST LOOK FOR " STEFANO GIORGINI " DIRTY DIANA"”
- "Step 1: Pause this video
  Step 2: Google 'Rainymood'
  Step 3: Click the first link
  Step 4: Unpause this video
  Step 5: Thumbs? up this comment, enjoy and thank me later",
- "Those 3,175 haters listen to? 'Techno'. "

The neural network was then trained and it was used to filter the input. After a 500-epochs training stage, the network was evaluated and the cross-validation results are presented in Table 1. This shows that the proposed method offers a good accuracy for removing irrelevant/spam comments. Moreover, using this classifier about 30% of the comments retrieved from YouTube for each video are discarded, thus not feeding them to the relevance scoring stage that is much more computational intensive. This stage is described in the next section.

TABLE I. RESULTS OF THE PRE-CLASSIFICATION TASK

Type of Instances	No. Instances	%
Correctly Classified Instances	174	87.46
Incorrectly Classified Instances	26	12.54
Total Number of Instances	200	-

V. RELEVANCE SCORING STAGE

A. Initial Approach

The initial approach had two main steps. The first step in the process was extracting the topics from the comments via the method presented above (using the Mallet library). After the topics were extracted, by using WordNet (http://wordnet.princeton.edu/), the number of topics was increased by adding synonyms and hyponyms of the topics extracted earlier. This was done to ensure that noise would not affect the final score. Because scores are given on a match basis without extracting synonyms and hyponyms comments with the same semantic composition, but using different terms to describe the same notion would receive different scoring results.

The second step consisted in fetching the appropriate Wikipedia article. This was done by performing a Google search using the wildcard *site:wikipedia.org* (to only obtain results from the wikipedia.org domain.) and the song author and title as the query. This, in most cases lead to the article corresponding to the respective song. The article was then processed as to remove any HTML entities such that only the bulk of the text is used in relevance scoring.

After the two steps were accomplished scoring would be given based on two features:

- Number of topics extracted
- Wikipedia article matches

B. *Topic-based Scoring*

The second method used for relevance scoring implies using the same two step approach as in the initial method but proposes a better approach for the second step. Topics extracted from comments are treated in the same way and are also enriched by using synonyms and hypernyms just as before, but for the second step, instead of using the entire wikipedia.org article text we attempted using only topics extracted from the respective Wikipedia article.

Topic extraction from the Wikipedia article was performed by using the same Mallet library as that used for extracting topics from the user comments but with different parameters as to ensure that enough of the topics are extracted from the text. The topics were also enriched with synonyms and hypernyms by using WordNet and after that the relevance scoring was performed. This method was preferred because it will remove noise from topics that consist of trivial words that would artificially increase the final score.

In this case the two features that provided the score were:

- Number of topics extracted
- Wikipedia topic matches

C. *Multiple Sources Topic-based Scoring*

As we presented before, initially we considered scoring by only using the Wikipedia article as a reference, but this might not always be the right way, especially for short Wikipedia articles related to the artist or the music composition. Also, using the full text as a reference was discarded and replaced with topic reference because the first could introduce a lot of noise due to commonly used words that would add up to the relevance score.

A quick solution to the problems above seemed to be using the Wikipedia references as additional resources with which to compute the relevance, but after some analysis of the idea we came to the conclusion that this procedure would not be reliable because, given the open-format of Wikipedia articles, some of them don't provide references or, in the fortunate case they provide them, they might refer to books or other documents that could not be automatically retrieved and analyzed. So instead of doing this we added three more sources, one of them being a popular music review website: http://www.allmusic.com , another being a different Wikipedia article and the last one would be the lyrics of the song at hand.

The first one comes to reinforce the information retrieved from Wikipedia and to ensure that relevant information present in both sources would get a boost in the relevance score. The second source, the lyrics help in increasing the score of persons quoting lyrics in the comments but their influence in the final score is lower as to not achieve very high ratings for comments that only contain lyrics.

As said before, full text reference was discarded and instead topics were extracted from both text resources. A total set of at least 80 topics was thus obtained, the size of the topic set depending on textual size of the articles found. The topics were extracted using the same Mallet library that was used to extract the topics from the comments but with adjustments for the training parameters. Because the intersection between the comment topic set and the reference topic set must be encouraged to increase the probability of obtaining a good measure of relevance, the reference topic set was also expanded by adding their synonyms and hyponyms to the set.

The second Wikipedia article is the article that provides information about the artist or artists that performed the specific piece of music. The name of the performers was parsed from the title of the video and a similar method as the one used before was employed to obtain the relevant topics from the article.

The lyrics were used for simple lexical matching as they already represent a non-trivial form of text, with a small bag of words that compose the whole text. The function that computes the relevance counts the occurrence of the topic terms in the article and if a term occurs in the topic extracted from the articles it increases the score.

The relevance score is also a function of the number of topics initially extracted. This increases the relevance of comments that contain multiple topics, or simply comments that have more content. Because the number of topics extracted from a comment was limited to 5 the influence of this feature on the total score was limited, as to maintain a low score of long comments that were not related to the respective video.

Finding the final relevance score is based on a weighed sum of scores obtained by matching the input topics to the three resources at hand. Each of the sources has a certain weight associated to the count of topics matched. The allmusic.com resource awards more points per topic match as articles are shorter and rich in information meaning that the chances of scoring on a trivial topic is reduced and there would be no significant relevance added to insignificant content.

VI. RESULT COMPARISON

In this section we will compare the results obtained by using the three methods of relevance scoring and attempt to establish which of them represents the most suitable one for rating comment relevance. The comparison has been done using test instances picked because they are considered very representative for the problem we want to solve. Such examples are:

- The Beatles- Here Comes The Sun[1]
- The Police - Every Breath You Take[2]
- Phil Collins - Another Day In Paradise[3]

These were chosen because both the artists and songs are very well known pieces of music and their value has been recognized worldwide. For the testing purposes we fetched the first 100 comments for each video. The comments were then purged using the pre-filtering phase presented above, some of them were removed and we were left with a slightly smaller corpus of comments for the classification phase.

[1] http://www.youtube.com/watch?v=U6tV11acSRk
[2] http://www.youtube.com/watch?v=OMOGaugKpzs
[3] http://www.youtube.com/watch?v=Qt2mbGP6vFI

The commentaries without any re-ranking applied can be found on the YouTube website as presenting them in the current paper would not be relevant due to the fact that they are just sorted descending by added date. Therefore we continue by providing the 5 most relevant comments for one the test instances (The Beatles- Here Comes the Sun). It can be easily seen that the second ranking – with results presented in Table 3 – and the third ranking – with results presented in Table 4 – methods that consider important the topics computed from the Wikipedia pages related to the title of the video as having a higher relevance provide very good ranking results, with more useful information for the interested user in good content related to the topic of the video. The results of the first method from Table 2, show that although it is simpler and less intensive computationally, it provides only adequate results which are less relevant than the ones offered by the other two methods.

TABLE II. RANKING OFFERED BY THE INITIAL APPROACH

Rank	Comment text	Relevance score
1	my mom said she doesn't like the beatles and she said? that john was only good to look at not to hear. my dad said, " haha so true!." i'm an orphan now.	123
2	because it's a fight, people? like you are against it. fights do not have to be violeny. our words are our weapons. has relevance	119
3	love this song...4 35 yrs? Now	114
4	this song is over 50? years old now! and it's still 10 times better then most "new music" today!	114
5	for the 1,435 dislike people. i hope you burn in? hell!!!	111

TABLE III. RANKING OFFERED BY THE TOPIC-BASED SCORING

Rank	Comment text	Relevance score
1	i didn't mean fight other places. i meant focus on the hurt people in your own country first, then expand to? the others. if people don't agree with peace that's an opinion. not a fact, and people often take offense to opinions. there isn't anything to take offense to, they say something that's all it is. they said it, don't put meaning to it. world peace - i meant the whole world having peace there.	1539
2	hey i know u just wanna listen to the song but i still have to write this hoping someone will see it and that someone will care .i'm? a young musician from croatia so this spam is my only chance to get noticed.please check out my channel and i promise u won't be sorry.i appreciate your time because music means everything to me, thank you!	1339
3	my mom said she doesn't like the beatles and she said that john was? only good to look at not to hear. my dad said, " haha so true!." i'm an orphan now.	1083
4	you shouldn't be listening to the beatles since these? seem to turn your friends into enemies! beatles are all about peace! you are not getting their message!	948
5	because it's a fight, people like you are against it.? fights do not have to be violeny. our words are our weapons.	666

TABLE IV. RANKING OFFERED BY THE MULTIPLE SOURCES TOPIC-BASED SCORING

Rank	Comment text	Relevance score
1	i didn't mean fight other places. i meant focus on the hurt people in your own country first, then expand to? the others. if people don't agree with peace that's an opinion. not a fact, and people often take offense to opinions. there isn't anything to take offense to, they say something that's all it is. they said it, don't put meaning to it. world peace - i meant the whole world having peace there.	1693
2	hey i know u just wanna listen to the song but i still have to write this hoping someone will see it and that someone will care .i'm? a young musician from croatia so this spam is my only chance to get noticed.please check out my channel and i promise u won't be sorry.i appreciate your time because music means everything to me, thank you!	1309
3	you shouldn't be listening to the beatles since these seem to turn your friends into enemies! beatles are all about peace!? you are not getting their message!	983
4	my mom said she doesn't like the beatles and she said that john was only good to look at? not to hear. my dad said, " haha so true!." i'm an orphan now	968
5	maybe your friend should know that being english, have a picture in abbey road and "sing" all you need is love" won't make one direction? a group like the beatles...	662

To finish with, we have also computed Spearman's rank correlation coefficient for the first 100 comments found on the YouTube website for each video.

The comments extracted from the website are not correlated with any of the three ranking methods presented in this section as the highest correlation coefficient is merely 0.031 with method 1. A simple explanation for this is that there is no correlation between the relevance of a comment and the time they are added. Therefore sorting the comments descending by added date does not provide any correlation with the methods proposed in this paper that should rank them according to their relevance.

However, method 1 is also not correlated with the other two methods, having a rank correlation of just 0.001 with method 2. At last, methods 2 and 3 are better correlated compared to the other ones, achieving a rank correlation of 0.124. This can be explained as the third method is just a refinement of the second one, thus we find a good correlation coefficient between them.

VII. CONCLUSIONS

Text classification is not a trivial problem, especially when dealing with small pieces of text, such as user comments. The approach presented in this paper for ranking comments according to their relevance for the video proved to be a good one, although it has its limitations. For example, it is applicable only to comments written in English and performs poorly on lesser known songs/artists.

The algorithm was tested on multiple videos from YouTube and a sample of the output is presented in the previous section, showing that the results are clearly more relevant than the ones presented default by YouTube. We picked a few popular videos to ensure that the rating system provides good results.

Furthermore, limiting the number of retrieved comments to 100 for each video still managed to highlight interesting comments for each video item.

Finally, we proved that the proposed two-step method for solving the problem can be used to build a good relevance ranking tool for comments on YouTube. The first step consists of pre-filtering the input via a neural network binary classifier, and the second step combines topic extraction with a weighted function relevance computing stage.

ACKNOWLEDGMENT

The research presented in this paper was partially supported by project No. 264207, ERRIC-Empowering Romanian Research on Intelligent Information Technologies (FP7-REGPOT-2010-1).

REFERENCES

[1] F. Sebastiani, "Machine learning in automated text categorization," ACM Comput. Surv., vol. 34, no. 1, 2002, pp. 1-47; DOI 10.1145/505282.505283.

[2] K. Lee, et al., "Uncovering social spammers: social honeypots + machine learning," Book Uncovering social spammers: social honeypots + machine learning, Series Uncovering social spammers: social honeypots + machine learning, ed., Editor ed.^eds., ACM, 2010, pp. 435-442.

[3] R. Malouf and T. Mullen, "Taking Sides: User Classification for Informal Online Political Discourse," Internet Research, vol. 18, no. 2, 2008, pp. 177-190; DOI citeulike-article-id:5722689.

[4] G. Inches and F. Crestani, "Overview of the international sexual predator identification competition at PAN-2012," Proc. CLEF 2012 Evaluation Labs and Workshop — Working Notes Papers, 2012.

[5] S. Siersdorfer, et al., "How useful are your comments?: analyzing and predicting youtube comments and comment ratings," Book How useful are your comments?: analyzing and predicting youtube comments and comment ratings, Series How useful are your comments?: analyzing and predicting youtube comments and comment ratings, ed., Editor ed.^eds., ACM, 2010, pp. 891-900.

[6] A.E.S. Baccianella and F. Sebastiani, "SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining," Book SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining, Series SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining, ed., Editor ed.^eds., European Language Resources Association (ELRA), 2010, pp.

[7] A. Ammari, et al., "Identifying relevant youtube comments to derive socially augmented user models: a semantically enriched machine learning approach," Book Identifying relevant youtube comments to derive socially augmented user models: a semantically enriched machine learning approach, Series Identifying relevant youtube comments to derive socially augmented user models: a semantically enriched machine learning approach, ed., Editor ed.^eds., Springer-Verlag, 2012, pp. 71-85.

[8] C.-F. Hsu, et al., "Ranking Comments on the Social Web," Book Ranking Comments on the Social Web, Series Ranking Comments on the Social Web, ed., Editor ed.^eds., IEEE Computer Society, 2009, pp. 90-97.