

Soodamani, R (2016), Metadata Extraction and Classification of YouTube Videos Using Sentiment Analysis, *ICCST2016*, Paper presented at the IEEE International Carnahan Conference on Security Technology, 24-27 October 2016, Orlando, Florida.

Metadata Extraction and Classification of YouTube Videos Using Sentiment Analysis

Shanta Rangaswamy, Shubham Ghosh, Srishti Jha
Department of Computer Science and Engineering
R.V. College of Engineering, Bengaluru, India

Soodamani Ramalingam
School of Engineering and Technology
University of Hertfordshire, Hatfield, UK

Abstract: MPEG media have been widely adopted and is very successful in promoting interoperable services that deliver video to consumers on a range of devices. However, media consumption is going beyond the mere playback of a media asset and is geared towards a richer user experience that relies on rich metadata and content description. This paper proposes a technique for extracting and analysing metadata from a video, followed by decision making related to the video content. The system uses sentiment analysis for such a classification. It is envisaged that the system when fully developed, is to be applied to determine the existence of illicit multimedia content on the web.

Keywords: MPEG, Metadata extraction, Video processing, sentiment analysis, polarity.

I. INTRODUCTION

Recently, there has been a series of terrorist attacks such as the July 2016 attack on Munich in Germany, Nice in France and Dhaka in Bangladesh, to name just a few. The gunman in the Nice attack is believed to have visited websites that showed pictures of executions before making his attack. Such terrorist activities call for monitoring of web activity related to the gunmen involved. Governments, in dealing with actions against such hate crime, have a need to censor such websites. However, the growth of the information uploads over the World Wide Web is increasing on a very rapid scale that it is impossible to sieve through them manually. Hence, there is a need for an automatic content analysis that that can listen to, read and extract relevant information that it is looking for which is termed as 'metadata'. There are many web platforms that are used to share non-textual content such as videos, images and animations that allow users to add comments for each item [1].

Sentiment analysis or opinion mining is one of the great accomplishments of the last decade in the field of Language Technologies. This field of study is related to the analysis of opinions, sentiments, evaluations, attitudes, and emotions of users which they express on social media and other online resources. The revolution of social media sites has also attracted the users towards video sharing sites. YouTube is probably the most popular of them, with millions of videos uploaded by its users and billions of comments for all of these videos. Online users express their opinions or

sentiments on the video that they watch on such sites. Classification of video is an increasingly prominent area of research, rising with the quantity of videos shared online through such sites [2-3]. In general, sentiment analysis attempts to determine the attitude of material contributors with respect to the topic of interest or the overall contextual polarity of the content. That is, whether the expressed opinion in the content may be classified for example as positive, negative or neutral or equivalent.

Thus, we can perform a sentiment analysis related to hate crime that would potentially extract related video instigating such terrorist activities in real life.

Literature review reveals that sentiment analysis has been typically been carried on textual data such as in [5]. Such systems involve carrying out a geographic analysis of crime data in understanding high crime areas and hot spots using Twitter data. Sentiment statistics enables the categorization of tweets by type, occurrences and number of associated tweets. A sentiment score indicating the central idea of tweets is determined. Sophisticated machine learning algorithms such as Deep Learning and Affective Computing techniques have been used for real-time analysis.

However, carrying out a similar task on video poses high volumes of data handling and extracting meaningful information from the video content, which is a non-trivial problem. This area of research is gaining importance due to the advancements and availability of automation tools [6].

Sentiment analysis on generic video content consists typically of the following stages [7]:

- Event classification- to classify an event with importance levels such as critical, high, etc. Automation tools such as Wordnet [8] may be used.
- Polarity detection-rating the opinion into positive, negative, neutral descriptions. Again tools such as Wordnet may be used. Furthermore, typical phrases used in social media may be interpreted to determine the polarity. For example, flagged, self-promotion, propaganda, abusive, etc. are some

typical phrase descriptions used in polarity extraction.

- Polarity prediction-Analysing users' comments for predicting polarity. This is related to deciding comments from viewers of videos and thereby deducting the polarity. Clustering and aggregation techniques may be used for this purpose.
- Evaluation of retrieved content based on metadata: Precision and Recall are two objectives measures commonly used in content based multimedia retrieval systems that is applicable here.

To the best of our knowledge, there is a lack of information on video metadata extraction on crime related content. This paper therefore proposes an initial model development that would effectively extract meta data from YouTube social media that is related to crime.

The rest of the paper is organized as follows: Section II is about the stages of the algorithmic approach of metadata extraction. Section III talks about the experimental analysis with different case studies, followed by conclusion and further work in section IV.

II. METADATA EXTRACTION – ALGORITHMIC APPROACH

The proposed system utilizes certain aspects of metadata based retrieval system such as in [10] for extracting the metadata within the video content. The general stages of development are described in the following sub-section.

A. Metadata Extraction- Stages of Development

The key stage involves understanding of the relevant database followed by aggregation of collected metadata information from selected website. The system architecture flowchart is illustrated in Fig.1.

1) *Metadata categorization*: The Dublin Network Working Group separates metadata into three groups namely content, intellectual property and instantiation and provides detailed description of each of their elements [9]. This classification is reproduced in Table I. Such a categorization is a useful step during the development of a metadata extraction process.

Table I Metadata classification [9]

Content	Intellectual Property	Instantiation
Title	Creator	Date
Subject	Publisher	Format
Description	Contributor	Identifier
Type	Rights	Language
Source		
Relation		
Coverage		

2) *Metadata extraction* from a media file involves retrieving information such as the name of the media file,

links to it, metatags, partner information supply the content, etc. The result of extraction is a list comprising of media and web page information such as URLs, titles, keywords, author, genre, etc. A typical output may appear as shown in Table II.

Table II Extracted Metadata

Field	Contents
The referring URL	http://www.youtube.com
Media URL	https://youtu.be/h0SXO5KUZIo
Title	"Cyber-security and Cyberwar: What Everyone Needs to Know"
Channel	Talks at Google

3) *Metadata parsing*: Some of the extracted data may often contain outliers such as noise, white spaces or figure of speech which is parsed and indexed, and has been cross checked against a database with set fields. This provides an opportunity to rectify and segregate the noisy fields, thereby adding the relevant data to the extracted list as shown in Table III.

4) *Metadata Lookup*: The meta data aggregated from the previous step is matched against a known database. Let us assume that a media file is found to have information related to "cyber security and cyber war" in a google talk by a media personality. The metadata is extracted and has the following fields as shown in Table III. The fields are compared against the known database. This process enables to identify if the website is related to promoting crime.

Table III Metadata parsing

Field	Contents
Published on	Feb 10, 2014
Title	"Cyber Terrorism and Warfare: The Emergent Threat"
Category	News and Politics
Number of views	Around 34k
Performer	Peter Warren Singer
License	YouTube Standard license

B. Metadata Extraction Principle and Procedure

The proposed system is developed in Python, version 2.7.9. Firstly, the source code of a YouTube webpage for a specific video (given its URL) is obtained using Python library urllib2 which uses **request.get ()** function. For metadata extraction, the Regular Expressions Python inbuilt library is used as this makes the process of string matching easier and efficient. Using string matching the Title, Description, Number of Views and Category of the video are extracted. Due to the flexibility in the metadata extraction code, any other information can be extracted. Further, the code uses Python's inbuilt Natural Language Processing Toolkit (NLTK), which performs text parsing of the metadata obtained above and it maps the required tags depending on nature of the word to each token. The code also considers a customized Python inbuilt Dictionary Corpus which consists of all the possible English words with their assigned ratings ranging from -15 to 15. This is then mapped with the required tags to calculate the aggregated

rating of the video. The above setup is run on Ubuntu 16.04 platform, but it can be made to run on a windows environment as well.

C. Metadata Extraction Implementation - Algorithm

In this Section, an algorithmic approach is outlined for metadata extraction and analysis of the extracted data. It works on the principle of seeking a URL, specifying specific fields of interest that form part of the query, parsing retrieved data, and classifying with reference to a predefined dictionary.

Step 1: Extraction of source code from a YouTube video

Fetch the source code of a YouTube webpage of a video given its URL using Python interface function `urlopen`:
`usock = urllib2.urlopen(url)`

The output is stored in `usock` and in the variable name is `"data"`.

Step 2: Extraction of specific elements of the video

Using python string manipulation operations match the string `"<meta itemprop="genre" content=""` from the inspect element of the video URL to the obtained source code. After the match occurs we copy the text following the matched string into a text file called `"Meta_data_outputs.txt"`. Pseudocode A provides the steps involved:

Pseudo Code A: Video information extraction

```
s2 = '<meta itemprop="genre" content=""  
y = data.index(s2)  
l = len(s2)  
s3 = '<div id="watch-header" class="yt-card yt-  
card-has-padding">  
y2 = data.index(s3)  
category = data [y+1: y2]  
category =re.sub (">', ", category)  
text_file = open ("Meta_data_outputs.txt", "a")  
text_file.write("\r Category:%s\r" %category )  
text_file.close()
```

The above procedure can be used to extract information such as the video description, number of views, likes and dislikes. Also, the comments can be extracted. Any other information which may help in classifying the video may also be extracted at this stage.

Step 3: Defining a dictionary of words

The next step is to recognize positive and negative expressions. This is done by referencing a customized dictionary file that is but with words and their assigned rating. Such type of customized dictionary file is named `"Corpus.txt"` and a sample from the same is shown below:

```
nice: [positive]  
motivation: [positive]  
inspirational: [positive]  
bad: [negative]uninspired: [negative]  
expensive: [negative]
```

This dictionary needs to be enhanced progressively for better results by addition of more words with their respective ratings.

Step 4: Parsing retrieved data

The retrieved data is in the form of a text as a series of sentences. These are stored in a file `Meta_data_outputs.txt`. During parsing of the output file, each token is assigned a specific tag using the NLTK. These tags are labels associated to each word depending on the type of the word, such as a noun, verb, adjective etc.

The following structure will be used:

- Each text is a list of sentences.
- Each sentence is a list of tokens.
- Each token is a tuple of two elements: a word form (the exact word that appeared in the text) and a list of associated tags.

Pseudo Code B: Metadata Parsing

```
tokens=nltk.word_tokenize(temp) # variable temp holds  
each sentence temporarily tagged=nltk.pos_tag(tokens)
```

Let us consider an example of retrieved text:

`"All that is gold does not glitter"`. This is tokenized as follows: `[['All', ['DT']], ('that', ['DT']), ('is', ['VBZ']), ('gold', ['NN']), ('does', ['VBZ']), ('not', ['RB']), ('glitter', ['VB']), (',', ['.])]`,

Another example: `"Not all those who wander are lost."`, this is tokenized as per the following:

```
[['Not', ['RB']], ('all', ['DT']), ('those', ['DT']), ('who', ['WP']), ('wander', ['NN']), ('are', ['VBP']), ('lost', ['VBN'])]
```

Step 5: Mapping Metadata to the Dictionary:

In this step, the list of required tags is determined as indicated in Pseudocode C.

Pseudo Code C: Tag Mapping with Rating

```
required tags=['JJ', 'JJR', 'JJS', 'NN', 'NNS', 'NNP', 'NNPS', 'RB', 'RBR', 'RS', 'VBG', '-NONE-', 'VBZ']
```

The mapped metadata has a predefined rating for the words present in the customized `"Corpus.txt"` dictionary file.

Step 6: For Calculating rating of the video:

General formula:

- The positive and negative ratings are added separately. Then final rating is calculated using an aggregation as shown in Pseudocode D:

Pseudo Code D: Rating Aggregation

```

tsum=psum-nsum
if sum>0:
    pol = (float) (tsum / psum)
    if pol!= 1:
        pol = pol * 0.5
        tot += pol + 0.5
    else:
        pol=div(max(pword),4) *0.5
        tot=tot+pol+0.5
else if sum< 0 :
    pol = (float) (tsum / nsum)
    if pol != 1 :
        pol = pol * 0.5 * -1
        tot += pol + 0.5
    else:
        pol=div(min(nword),-4)*(-0.5)
        tot=tot+pol+0.5

```

Here, **pol** is a temporary variable, and **tot** is the total rating of the document.

- Number of sentences are calculated. Average rating of entire document is calculated using:
`if sentence! =0: rating=(tot/sentence)`
 where "sentence" is number of sentences.

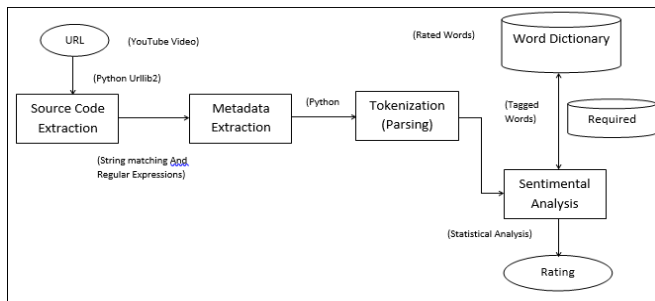


Fig.1: Diagrammatic representation of the entire implemented system.

III. EXPERIMENTAL ANALYSIS

In this Section, we consider a case study that demonstrates the principle of metadata extraction from video and its classification. Firstly, the datasets used for metadata extraction is described followed by an analysis of the results obtained for the datasets.

A. Dataset

The dataset consists of a set of 15 YouTube videos. The dataset categorization is as follows:

- Negative Sets: A set that is inappropriate for viewership as they deal with criminal cases such as theft, harassment and abuse.
- Positive Sets: A set of videos having high viewership content and dealing with motivational and inspirational values are considered. These

videos also talk about security and safety related issues.

- Neutral Sets: A set of videos which are neutral and don't have any impact (positive or negative) on the society. So each video is given a rating which ranges between 0 to 1, based on the algorithm designed above. Depending on the above rating videos are classified into three class labels: Positive, Negative and Neutral.

Case Study: Let's consider the URL "<https://www.youtube.com/watch?v=Tdv3TiuFvMs>", The metadata contained within the URL is as following:

Title: One Minute Inspiration - Never Give Up! Be Successful!

No of views: 25

Published on: Apr 18, 2014

Success is just around the corner for those who seize the day and refuse to give up. Your goals are important and can be met if you try hard and persevere.

You can do it!

Category: People & Blogs

License: Standard YouTube License"

This metadata is extracted and stored as a text file: "*Metadata_output.txt*".

This output file is then parsed to obtain the following tokens:

```

< ['\\Title', 'One', 'Minute', 'Inspiration', 'Never', 'Give', 'Up',
'Be', 'Successful', 'YouTube', 'Category', 'People', 'amp',
'Blogs', 'Find', 'out', 'why', 'Close']
[No', 'of', 'views', '26', 'Description', 'Published', 'on', 'Apr',
'18', '2014', 'Success', 'is', 'just', 'around', 'the', 'corner', 'for',
'those', 'who', 'seize', 'the', 'day', 'and', 'refuse', 'to', 'give', 'up']
[Your', 'goals', 'are', 'important', 'and', 'can', 'be', 'met', 'if',
'you', 'try', 'hard', 'and', 'persevere']
[You', 'can', 'do', 'it', '!'] >

```

The next process is to obtain all possible tags from the natural language processing toolkit (NLTK) which are required for calculation of rating:

Required tags= ['JJ', 'JJR', 'JJS', 'NN', 'NNS', 'NNP', 'NNPS', 'RB', 'RBR', 'RS', 'VBG', '-NONE-', 'VBZ']

Tokens that are obtained from the tokenized metadata and the tags obtained above are mapped together to associate a predefined rating for the words present in the customized dictionary.

Example illustrated below:

#####

[('Title', 'NN'), ('One', 'CD'), ('Minute', 'NNP'), ('Inspiration', 'NNP'), ('Never', 'RB'), ('Give', 'VBP'), ('Up', 'RP'), ('Be', 'NNP'), ('Successful', 'JJ'), ('YouTube', 'JJ'), ('Category', 'NN'), ('People', 'NNP'), ('&', 'CC'), ('amp', 'NN'), ('Blogs', 'NNP'), ('>', 'NN'), ('Find', 'NNP'), ('out', 'RP'), ('why', 'WRB'), ('Close', 'JJ'), ('No', 'NN')]

('word = ', 'inspiration');
('rating = ', 10)
('word = ', 'Close');
('rating = ', 2)
('word = ', 'Successful');
('rating = ', 8)

#####

[('No', 'NN'), ('of', 'IN'), ('views', 'NNS'), ('26', 'CD'), ('Description', 'NN'), ('Published', 'VBN'), ('on', 'IN'), ('Apr', 'NNP'), ('18', 'CD'), ('2014', 'CD'), ('Success', 'NNP'), ('is', 'VBZ'), ('just', 'RB'), ('around', 'IN'), ('the', 'DT'), ('corner', 'NN'), ('for', 'IN'), ('those', 'DT'), ('who', 'WP'), ('seize', 'VBP'), ('the', 'DT'), ('day', 'NN'), ('and', 'CC'), ('refuse', 'NN'), ('to', 'TO'), ('give', 'VB'), ('up', 'RP')]

('word = ', 'just');
('rating = ', 3)
('word = ', 'Success');
('rating = ', 7)

#####

[('Your', 'PRP\$'), ('goals', 'NNS'), ('are', 'VBP'), ('important', 'JJ'), ('and', 'CC'), ('can', 'MD'), ('be', 'VB'), ('met', 'VBN'), ('if', 'IN'), ('you', 'PRP'), ('try', 'VBP'), ('hard', 'JJ'), ('and', 'CC'), ('persevere', 'JJ')]

('word = ', 'persevere');
('rating = ', 5)
('words = ', 'goals');
('rating = ', 8)

#####

The aggregated rating of the entire set of extracted meta-data is calculated using the algorithm explained in section II (Algorithmic approach- Step 6) and will be categorized based on the two threshold values defined in the next section.

Result of the above sample set: **Overall rating of the video is: 0.9375**. The ratings for the whole dataset is given in Fig.2.

Category: Positive

B. Metadata Extraction: Sensitivity Analysis

The classification of the videos is done based on the final aggregated rating obtained. A range of ratings have been defined that is used to classify the videos as positive, negative or neutral. Each of these set of ranges have a defined threshold. We analyse the performance by using two different thresholds:

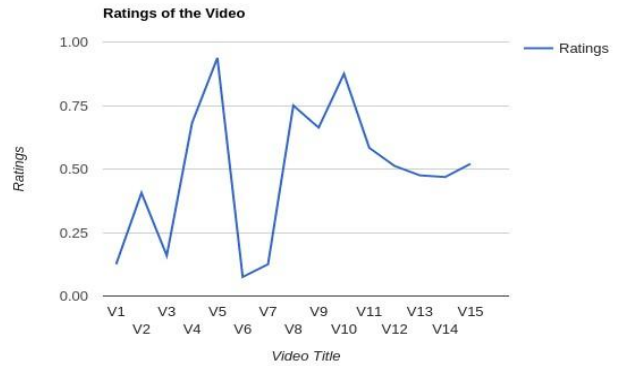


Fig. 2. Graph Depicting Video Number Vs Rating

Table IV: Use of thresholds for determining polarity

Video No.	Threshold 1	0 to 0.3 Negative	0.3 to 0.7 Neutral	0.7 to 1.0 Positive
	Threshold 2	0 to 0.45 Negative	0.45 to 0.65 Neutral	0.65 to 1.0 Positive
Overall Rating				
V1	0.125	✓		
		✓		
V2	0.40625		✓	
		✓		
V3	0.1591	✓		
		✓		
V4	0.6819			✓
				✓
V5	0.9375			✓
				✓
V6	0.075	✓		
		✓		
V7	0.125	✓		
		✓		
V8	0.75			✓
				✓
V9	0.6629			✓
				✓
V10	0.875			✓
				✓
V11	0.5834		✓	
			✓	
V12	0.512		✓	
			✓	
V13	0.475		✓	
			✓	
V14	0.4685		✓	
			✓	
V15	0.5205		✓	
			✓	

For example, Threshold 1 is set as: (0 to 0.3) - Negative; (0.3 to 0.7) - Neutral; (0.7 to 1.0) - Positive.

Similarly, Threshold 2 is set as: (0 to 0.45) - Negative; (0.45 to 0.65) - Neutral; (0.65 to 1.0) - Positive.

These thresholds then classify the videos into their respective class labels. The resulting classification of video dataset is provided in Table IV.:

From Table IV, it can be observed that though the threshold values set for negative, neutral and positive changes; the category into which a video is classified doesn't change by much. The dataset used in the experimental analysis includes 15 videos each ranging from 30MB in size to 400MB. Also, the metadata extracted such as description, number of views, and category of a video is a text file and has size in KB. But, when the size of the dataset increases the classification will also start varying. The number of variations within each threshold will be large which is not seen with the above sample range. It has also been observed that threshold 2 classifies each of the videos accurately into either positive negative and neutral. Hence, threshold 2 is preferred over any other boundary conditions for each of the class labels.

When Threshold 1 was used to classify the above sample set, one video was classified inaccurately while the other 14 were accurate. On the other hand, with threshold 2 all of the 15 videos from the above sample set were classified correctly. This variation in result is due to this particular sample set size. It is believed that if the size of the dataset increases the classification will also start varying. For a larger dataset, different threshold ranges will need to be established for much better classification.

IV. CONCLUSION AND FURTHER WORK

In this paper, a technique for analyzing YouTube video URLs is proposed. The technique follows a sentiment analysis to determine the polarity for the video objects. The system has been tested on a small dataset. The results are promising.

The accuracy of the entire process depends on the list of words with their respective ratings which are present in Python inbuilt dictionary Corpus. For the purpose of testing the enhancement of Corpus, by adding new words and their respective ratings was done manually. This process needs to be automated with growing size of the dataset. A manual

update of the Corpus dictionary would become time consuming and impractical. To automate the process, we propose using Machine Learning concepts such as Neural Networks, Genetic Algorithms, SVMs and Bayesian Learning. Further, when the size of the dataset increases the classification of the videos may vary and in order to have a robust classification, different threshold ranges will need to be established.

V. REFERENCES

- [1] Choudhury, Smitashree and Breslin, John G. (2010), "User sentiment detection: a YouTube use case", Proc. The 21st National Conference on Artificial Intelligence and Cognitive Science, 30 August - 1 September 2010, Galway, Ireland, 2010.
- [2] Kundi, F. M., Ahmad, S., Khan, A., & Asghar, M. Z. Detection and Scoring of Internet Slangs for Sentiment Analysis Using SentiWordNet. Life Science Journal, Vol. 11 No. 9, 2014.
- [3] G. Vinodhini and RM.Chandrasekaran, "Sentiment analysis and opinion mining: a survey", International Journal of Advanced Research in Computer Science and Software Engineering, Vol.2, No.6, pp. 282-292, June 2002.
- [4] Jain, R. and Fuller, C. and Gorkani, M.M. and Horowitz, B. and Humphrey, R.D. and Portuesi, M.J. and Shu, C. and Hampapur, A. and Gupta, A. and Bach, J.}, "Video cataloger system with extensibility", Google patents, US Patent 6463444, 2002.
- [5] Raja Ashok Bolla," Crime pattern detection using online social media", MSc Thesis, Missouri University of Science and Technology, US, 2014.
- [6] Wootton, Cliff, "Developing quality metadata: building innovative tools and workflow solutions", Focal Press, 2009, ISBN-13:978-0-240-80869-7.
- [7] Muhammad Zubair Asghar, Shakeel Ahmad, Afsana, Marwat, Fazal Masud Kundi, "Sentiment Analysis on YouTube: A brief survey", MAGNT Research Report Vol.3, No.1, pp: 1250-1257, 2015, ISSN. 1444-8939.
- [8] Fellbaum, Christian, "WordNet and wordnets". In: Brown, Keith et al. (eds.), Encyclopedia of Language and Linguistics, Second Edition, Oxford: Elsevier, 665-670, 2005.
- [9] Stuart L. Weibel, John A. Kunze, Carl Lagoze and Misha Wolf, "Dublin Core Metadata for Resource Discovery", The Internet Society 1998, URL: <http://www.ietf.org/rfc/rfc2413.txt>, last accessed: 27 July 2016.
- [10] Ken Alan Berkun, Austin David Dahl, Jennifer Lynn Kolar, Scott Chao-Chueh Lee, Shannon E. McRae, Brad Steven, Miller, Mercer Island, John Prince, Bellevue, Eric Carl Rehm, Srinivasan Sudhanagunta, Seattle, Jonathan Robert Nowitz, Seattle, "Interpretive Stream Metadata Extraction", Patent No. US 2002/0103920 A1, 1 Aug.2002.
- [11] Siersdorfer, S., Chelaru, S., Nejdil, W., & San Pedro, J. "How useful are your comments? analyzing and predicting youtube comments and comment ratings", ACM. In Proceedings of the 19th international conference on World Wide Web, pp. 891-900, 2010.
- [12] Chetan Verma, Sujit Dey, Methods to Obtain Training Videos for Fully Automated Application-Specific Classification, IEEE Access, The Journal for rapid open access publishing, DOI: 10.1109/ACCESS.2015.2461156, Aug 7 2015, pp1188- 1205
- [13] Chng, X., Dale, C., & Liu, J., "Understanding the characteristics of internet short video sharing: YouTube as a case study", Technical Report arXiv 0707.3670v1, Cornell University, July 2007.

APPENDIX A: List of Videos used as dataset:

- How to Make a Bomb Cracker (Home Made) - Easy Tutorials - YouTube
(<https://www.youtube.com/watch?v=o--87j1dysU>)
- How to make Coloured smoke from Wax Crayons. Smoke bomb/ grenade for paintball, airsoft..etc. - YouTube(
https://www.youtube.com/watch?v=fdeXcGkqT_4)
- Thieves Stealing ATM - YouTube
(<https://www.youtube.com/watch?v=fGiCaheWGbI>)
- Video: Mom Ditches Baby at Walmart After Shoplifting - YouTube
(<https://www.youtube.com/watch?v=pgn3j1FHOVs>)
- Cyber Terrorism and Warfare: The Emergent Threat - YouTube
(<https://www.youtube.com/watch?v=CNE1tQoObbs&feature=youtu.be>)
- Video that will change your life. I have no words left. - YouTube (<https://www.youtube.com/watch?v=PT-HBI2TVtI>)
- STOP KILLING TIME ► Motivational Video - YouTube
(<https://www.youtube.com/watch?v=UX2tefQHNmk>)
- World Best Motivational Videos for Students - YouTube
(<https://www.youtube.com/watch?v=Tjinq5StX68g>)
- INSPIRATIONAL - HOW GREAT I AM - YouTube
(<https://www.youtube.com/watch?v=V6xLYt265ZM>)
- One Minute Inspiration - Never Give Up! Be Successful! - YouTube
(<https://www.youtube.com/watch?v=Tdv3TIuFvMs>)
- Factory made gasoline vapor carburetor from the past – YouTube
(<https://youtu.be/zaiygknSFHs>)
- Mark Zuckerberg 2016 The Lifestyles of Young Billionaire Entrepreneurs - YouTube
(<https://youtu.be/gB9Tv7vtWVU>)
- How to Fly a Drone
(<https://youtu.be/OcxUCepBHkM>)
- Playing to the Edge with General Michael Hayden - YouTube
(<https://youtu.be/etffkFDm2NQ>)
- Jeremy Howard: The wonderful and terrifying implications of computers that can learn - YouTube
(<https://youtu.be/t4kyRyKyOpo>)