How can users’ comments posted on social media videos be a source of effective tags?

Mehdi Ellouze

Abstract

This paper proposed a new approach for the extraction of tags from users’ comments made about videos. In fact, videos on the social media, like Facebook and YouTube, are usually accompanied by comments where users may give opinions about things evoked in the video. The main challenge is how to extract relevant tags from them. To the best of the authors’ knowledge, this is the first research work to present an approach to extract tags from comments posted about videos on the social media. We do not pretend that comments can be a perfect solution for tagging videos since we rather tried to investigate the reliability of comments to tag videos and we studied how they can serve as a source of tags. The proposed approach is based on filtering the comments to retain only the words that could be possible tags. We relied on the self-organizing map clustering considering that tags of a given video are semantically and contextually close. We tested our approach on the Google YouTube 8M dataset, and the achieved results show that we can rely on comments to extract tags. They could be also used to enrich and refine the existing uploaders’ tags as a second area of application. This can mitigate the bias effect of the uploader’s tags which are generally subjective.

Keywords

Video tagging · Social media comments · Self-organizing map · YouTube

Abbreviations

SOM Self-organizing map
GA Genetic algorithms
DTM Document term matrix
FS Frequency score
TF Term frequency
CF Comment frequency
TRECVID Text retrieval conference video retrieval evaluation
NGD Normalized Google distance

1 Introduction

The unprecedented spread of videos contents generated the need of tagging this content to make it easily browsable. Every second, thousands of videos are shared on the internet and social media by uploaders. Actually, we are facing a new important issue related to how to find out about the context and the story of the shared video. Some social media repositories like YouTube or Flickers or Instagram offer the uploader the possibility to tag the content when uploaded. However, in some others like Facebook, no tags are added to the video content. The only thing that we can do is to add a title to the video to help understand the original content. In the research community, important efforts are being made to create tools to allow exploring an important number of videos. Most research efforts focused on automatically recognizing semantic concepts in the video through computer vision and machine learning techniques. The semantic concepts can be entities, objects, events, and places. The TRECVID evaluation campaign supported these efforts for many years. TRECVID has provided researchers with a huge dataset made of annotated videos, by the means of which they train their visual concept detectors (Fig. 1).

However, the automatic concept detection systems suffer from some limitations. First, the number of concepts detectors is limited. Besides, the concepts contained in the original video should be known to be able to use the suitable detectors. Finally, the detectors reliability should be good enough
Fig. 1 Comments in video clips in YouTube

regardless of the quality of the original video. Moreover, there is a difference between tagging and concept detection. Concept detection means retrieving objects like a car, a person, a phone, a horse, an airplane, the sun, a forest, a tree, whereas tagging is somewhat different. For example, in a video showing Steve Jobs introducing the iPhone in 2007, the detected concepts might be a phone, a person, etc. However, the words that can properly describe the video are Steve Jobs, iPhone, 2007, etc. Named entities, places, dates, and circumstances are hardly detected by visual concepts detectors because they cannot always be extracted from pixels. This is what we usually call the “semantic gap.” The problem of video tagging has received less attention from the research community. On the contrary, many efforts have been made in image tagging. Three types of research works can be distinguished in the case of image tagging: tag relevance, tag refinement, and tag to region localization. In image tagging, the use of the social media has become more and more popular. Images on the social media like Flickers are tagged by their uploaders. When a new image needs to be tagged or the relevance of the existing tags are to be evaluated, we can use tags of images having similar content on the social media. Indeed, in social media the multimedia content (video, image, audio) is continuously enriched with information that can be used to understand the original content such as tags, titles, and comments.

In this paper, we focused on the problem of tagging videos shared on the social media. Most of these videos are untagged, which makes their browsing time-consuming. The core philosophy of social media is to support and promote collaboration between content creators and consumers (viewers). Comments are the tool used by viewers to express their feedback. A comment is a textual description of a viewer’s experience with the video. The analysis of video comments revealed that they may be used as a source of tags.

In this paper, we do not pretend that comments on social media are a perfect solution for tagging videos. We tried only to investigate the reliability of comments to tag videos. The question we would like to address is how comments can serve as a source of tags.

The motivation underlying our reason to launch on this work is to help in:

- Tagging videos on social media (Facebook, for instance),
- Enriching human made tags with other ones extracted from comments,
- Correcting and updating human made tags with objective ones since the added tags are generally personalized and subjective [1].

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3 introduces our contribution, whereas Sect. 4 details our approach. The obtained results of our approach are revealed and discussed in Sect. 5. The major conclusions are drawn in the final section before suggesting some future research perspectives.

2 Related work

Compared with image tagging, relatively fewer studies have been conducted on video tagging.

2.1 Tagging at video level

Siersdorfer et al. [2] propose an approach based on tags propagation. Tags assigned to a video on YouTube, for instance, are propagated to other videos using near duplicate techniques. If two videos share similar contents, tags from one video are transferred to the other.

Similar to [2], Zhao et al. [3] propose an approach based on tags propagation. However, their approach is rather more developed. First, they proposed a new approach for near duplicate videos techniques and their tags assignment technique is more refined. Candidate tags brought from similar videos are scored according to the level of similarity. Tags with high scores are then selected.

Viana et al. [4] propose an innovative approach for video tagging based on dealing with the tagging process as a game. A special application called Tags4VD is created, and users collaborate to tag videos. The purpose of tagging is to obtain a good score and win a reward. When tagging a video, the user is also asked to make sure that his proposed tags are
validated by the other players. In this way, the users validate the work of each other.

Mazloom et al. [1] propose to construct the TagBook representation of an unlabeled video by propagating tags from a large set of N socially tagged videos. They relied on the assumption that visually similar images shall have similar tags.

### 2.2 Tagging at (scene) shot level

Ballan et al. [5] present a review of the state of the art of data-driven methods for automatic annotation of social media. The authors particularly analyzed the nearest neighbor methods. These methods are based on propagating tags on the basis of visual similarities. The authors also presented a comparison of tag refinement methods for social images using standard datasets.

In another work, Ballan et al. [6] propose an approach for video retrieval based on tag propagation. In this work, the authors sought all the images from Flicker, Google, and Bing with similar tags and similar content. The video is segmented into shots, and key frames are extracted. The visual similarity is measured according to the key frames. The obtained tags from the retrieved images are added to the video and are assigned to key frames.

Baraldi et al. [7] propose a system which uses computer vision and machine learning techniques to tag videos. First, the authors segment the video into scenes. After that, the speech of the video is transcribed and processed to extract concepts. These concepts are mapped to visual features through machine learning techniques.

Zhu et al. [8] suggest to tag video shots by propagating tags coming from an already tagged video. They measure the similarity between a group of videos (training videos) and a test shot. Tags attached to videos are propagated to the test shots according to the similarity between them.

Wang et al. [9] propose an event video summarization system. They addressed the problem of video retrieval based on human-made tags. When looking for a video using keywords on YouTube, for instance, many videos are proposed and should be browsed to retrieve the sequence or the part matching the keywords inside each video. The proposed system tries to localize tags inside the video. Video shots are detected, and a score is given to each shot according to its importance. After that, the tags made on the video are assigned to each shot using an optimization model. When a user makes a query for a video sequence, the system tries to find the shots dealing with the text query in the target video and present the result as a video summary or a storyboard displaying only shots of interest.

Some works [10,11] propose to crawl twitter streams to extract game highlights, where hashtag peaks might indicate the most exciting moments in the game. The extracted hashtags are then used to tag video sport games.

### 2.3 Keyword extraction in social media

Chiraratanasopha [12] addressed the problem of keyword detection and sentimental analysis to analyze the feelings of Thai people in the COVID-19 epidemic from Facebook posts. To detect a keyword, they use a word segmentation and the classical TF-IDF technique. In the same way, Chou et al. [13] propose an automatic restaurant information and keyword extraction system. They analyze the blog posts data and extract the restaurant information such as name, address, and phone number they use the TF-IDF approach to extract hot keywords. Yang and Yang [14] use nearest neighbor blogs to improve the keyword extraction performance. The keywords extracted from the blog are not only those located in the blog, but also those coming from nearest neighbor blogs.

Marujo et al. [15] proposed a corpus of tweets annotated with keywords consisting of 1827 tweets. This corpus is used to automatically build and evaluate keywords extraction on Twitter.

Park et al. [16] propose to extract keywords of blogs, and they introduce the concept of richness of content. To be considered as a keyword, the blog should include the term and the related subtopics. For this reason, for each candidate keyword the related subtopics are detected throughout the Web; then, we should look for them inside the block to evaluate the effectiveness of the keyword.

Campos et al. [17] propose an unsupervised automatic keyword extraction method which works on statistical text features extracted from single documents to select the most relevant keywords of a text.

### 2.4 Contributions

Our contributions can be summarized as follows:

- We may say that this is a pioneer paper to introduce an approach to extract tags from comments posted on videos on the social media.
- It is based on the observation that comments may contain information describing the content of the video. We used natural language processing, statistical, and pattern recognition techniques to retrieve the desired tags.
- This study proposed an exploration algorithm able to mine the outputs of the self-organizing map to extract tags.
- The proposed approach was applied to YouTube 8M dataset, proposed by Google, which is one of the most important video datasets.
3 Proposed approach

Figure 2 shows an overview of the proposed approach. First, the comments were extracted from YouTube videos. After that, some treatments were performed on the comments to eliminate stop-words, special characters, emoticons, etc. Moreover, to be considered as a tag, the word should respect the following criteria:

- **Redundancy:** To be considered as a tag, the word should be redundant and repeated in several comments. It is some kind of voting of users which will make this word a tag.
- **Closeness:** To be considered as a tag, the word should be semantically close to other selected tags. We are looking for objective tags, which reflect principal events, people, locations, and concepts displayed in the video. For instance, if we have a video presenting the new features of a smartphone and in a part of the video a person is shown riding a bicycle talking on the smartphone, the bicycle could not be considered as tag for this video. To be considered as a tag, the word should reflect objectively the important elements and should faithfully represent the general idea of the video.

For this reason, after cleaning the comments, we filter the comments to retain only redundant words. We compute a score for each word, according to its redundancy in the comments. Only tags that have a high score are selected to be candidates as final tags. The last steps are the clustering and the mining. Words having important scores are clustered using self-organizing maps to detect agglomerations of tags. Finally, we explore the SOM [18]. The extracted tags are words that belong to large agglomerations.

3.1 Comments crawling

The goal of this module is the extraction of YouTube video comments. We are only interested in videos where the number of comments is important (exceeding 1000 in our case). In order to address this task, a crawler was implemented. It extracts video comments using the Web API through HTTP GET method [19].

3.2 Word filtering

3.2.1 Comments preprocessing

In this step, we carried out some preprocessing on these unstructured comments to generate the datasets. We should keep only words that could be used as tags. We applied the following preprocessing tasks: First, we performed a part-of-speech tagging [20] and then a data cleaning (removing stop words) using Laurence Anthony’s software [21]. Second, we removed all the expressions which were irrelevant for the proposed methodology like date (“Dec 2-2010” or “2-12-2010”), link (www.imdb.com, www.tmdb.com etc.), numbers (12, 20 etc.), special characters (“,” “,” “,” “,” “,” “,” “,” ”, “,” “,” “,” “,” “,” “,” ”, “,” “,” “,” “,” “,” “,” ”), emoticons and different languages (Chinese, Arabic, Bangla, Hindi, etc.).

3.2.2 Frequency scoring

The importance and effectiveness of a word to be an eventual tag, we should rely on two important criteria:

1. The number of times the word appears in each comment (Term Frequency [22]),
2. The number of comments that contain the word.

A good candidate word is the one which is frequently used in comments and which is included in the majority of the comments. We compute the document term matrix [23]. It counts how often each word is repeated in each comment (Fig. 3).

Based on the DTM, we compute a frequency score $FS$. It is a way to score the importance of words in the comments based on how frequently they appear within multiple comments. This score is high for words that have a high term frequency across the entire comments. The $FS$ score is the product of two statistics: Term frequency and comment frequency. Term frequency (TF) is the number of times the word occurs in comments. Comment frequency (CF) is the number of comments containing the word divided by the total number of comments.

$$TF(W) = \frac{\text{(Number of occurrences of } W \text{ in comments)}}{\text{(Total number of words in the comments)}}$$ (1)

$$CF(W) = \frac{\text{(Number of comments containing } W \text{)}}{\text{(Total number of comments)}}$$ (2)

$$FS(W) = TF(W) \times CF(W).$$ (3)

![Fig. 3 Document term matrix](image-url)
Frequent words are retained using the Pareto 20/80 selection algorithm. The distribution of scores according to the law of Pareto is that 80% of the scores are achieved by 20% of the words.

Table 1 displays our filtering process results on some videos of our evaluation dataset. For each video, we displayed the retained words labeled with their frequencies. We can notice that the effective tags can be easily distinguished. The clustering steps will help to isolate them.

### 3.3 Tags clustering

The retained tags after the filtering step may include some frequent words that are not semantically related to the video. The question is how to evaluate whether a given word is related to the video context or not. The comments generally include concepts, events, and names of people included in the original video. The clustering step aims at tracing a cartography of topics evoked in the comments. We would retain only related topics. For this reason, we should cluster the tags and evaluate the closeness of these clusters.

#### 3.3.1 Self-organizing map

The SOM was originally proposed by Kohonen [18]. It is based on the idea that systems can be designed to emulate the collective cooperation of the neurons in the human brain. Self-organizing maps (SOM) consist of neurons organized in a two-dimensional grid on which we can map input information. SOMs are able to learn and organize information. Semantically close inputs are mapped generally in close zones of the SOM. When learning, the SOM assigns each input element to a neuron of the SOM. Close inputs are assigned either to the same neuron or to adjacent neurons. The fundamental properties of SOM are:
– It maps high-dimensional data to two-dimensional data,
– It preserves the semantic relationships between input elements when they are mapped on the grid,
– It is noise tolerant.

Self-organizing maps have already been used for text clustering. For instance, in [24], the authors used the SOM for clustering change requests in industrial processes to make their treatment easier. In [25], the authors used the SOM for clustering words based on the morphological properties of the context words. The results are encouraging and show the ability of SOM to achieve words clustering efficiently. We have already used them to achieve a macro-segmentation. In our previous papers [26–28], we used the SOM to segment news broadcast programs and films into stories by discovering the clusters of video shots. They are well suited for mapping high-dimensional data (shots) into a two-dimensional space.

The self-organizing map algorithm is a competitive type of learning: During the data (entry) presentation to the network, neurons compete in such a way that only one of them, the “winner,” is finally active; it is the neuron whose prototype has the least difference with the data presented to the network. The principle of competitive learning is then to reward the winner. To this end, the weights of the connections are reinforced with the entries. The neuron having won the competition determines the center of an area of the map called neighborhood, whose extent varies over time. The next phase, called update (or adjustment), changes the position of the prototypes (link weights) to bring them closer to the data supplied to the network (Fig. 4).

The SOM algorithm may be summarized as follows:

1. Random initialization of weight links $W_{i,j}$,
2. Determination of the winner neuron by selecting the closest neuron $j^*$ to the input:

$$j^* = \arg\left(\min_j \sum_{i=1}^{n} (x_i(t) - W_{ij}(t))^2\right),$$  \hspace{1cm} (4)

3. Modification of the weight vector of the winner neuron at every iteration “t”:

$$W_{ij}(t) = W_{ij}(t-1) + a(t)[x(t) - W_{ij}(t-1)],$$  \hspace{1cm} (5)

4. Modification of neighboring neurons weight vectors of the winning neuron:

$$W_{ij}(t) = W_{ij}(t-1) + a(t)h(j^*, t)[x(t) - W_{ij}(t-1)],$$  \hspace{1cm} (6)

where $a(t)$ is the learning coefficient and $h(j^*, t)$ is the interaction function:

$$h(j^*, t) = \exp\left(-\frac{j - j^*} {2r^2(t)}\right).$$  \hspace{1cm} (7)

During the SOM learning, phases 2 through 4 are repeated many times until the weights become stable.

Contrary to other clustering techniques such as K-Means, the SOM also provides information on the clusters’ closeness thanks to the neighborhood feature. The advantage of SOM is the fact that we are not required to provide the total number of the final clusters. It will project words on a two-dimensional map, making the distance between close words on the map as small as possible. The selection of final tags will be easier because we will just select the most important clusters (Fig. 5).

### 3.3.2 Compute features

To cluster words, we should compute the features (a descriptor) for each word. The computed features should reflect the closeness of words. We need to quantify the semantic relationship between words to identify real video tags and skip others that are irrelevant. Two different similarity measures are used to quantify the semantic relationship. The first is the Leacock similarity measure. It is based on the linguistic closeness between words. The second is the normalized Google distance. It is based on the output of Google search engine. If the number of words is $N$, we compute, a $N\times N$ similarity matrix $S$ where the entry at position $(i, j)$ represents the similarity between the tag $i$ and the tag $j$. The $i$th column represents the descriptor for the $i$th word (Fig. 6).
3.3.3 Normalized Google distance

The normalized Google distance (NGD) is a semantic similarity measure computed through counting the number of hits returned by the Google search engine for a given keyword. The NGD is widely applied in different research studies [29–31].

Words which are semantically related are normally located in many common Web pages. To evaluate the NGD among different word pairs, we used the Cilibrasi and Paul formula [32]

$$\text{NGD}(x, y) = \frac{\max\{\log(f(x), \log f(y)) - \log f(x, y)\}}{\log(M) - \min(\log f(x), \log f(y))}.$$  \hspace{1cm} (8)

- $f(x)$ represents the return pages of Google search engine containing $x$.
- $f(y)$ represents the return pages of Google search engine containing $y$.
- $f(x, y)$ represents the return pages of Google search engine containing $x$ and $y$.
- $M$ represents the total Web pages that Google search engine obtains.

For instance, if we want to measure the NGD between “music” and “piano,” the total number of pages containing “music” is $f(\text{music})=2,760,000,000$. The return result of “piano” is $f(\text{piano})=186,000,000$. We input “music” and “piano” in the Google search engine, and the result is $f(\text{music}, \text{piano})=65,700,000$. After calculation, the NGD between “music” and “piano” is 0.74545835.

The smaller the NGD is, the shorter the distance between two words is and the greater the similarity will be. The NGD of two similar words is zero.

3.3.4 WordNet Leacock similarity

WordNet is the product of a research project at Princeton University [33]. It is a large lexical database of English. In WordNet nouns, verbs, adverbs, and adjectives are organized relying on a variety of semantic relations into synonymous sets (synsets). They represent one concept. Leacock similarity is a similarity measure based on WordNet. When considering the WordNet taxonomy, each node represents a unique concept (or synset) in the taxonomy. Subsequently, the similarity degree between a word pair is computed by calculating the shortest path between two concepts (represented as nodes) and dividing it by twice the maximum depth of the graph.

$$\text{LCH}(x, y) = \max\left[-\log \frac{\text{length}(x, y)}{2D}\right].$$  \hspace{1cm} (9)

3.4 Mining and exploring the SOM

When training the SOM [18], we fixed the size of the map using the formula of Vesanto et al. [34]. The number of neurons is computed as $5\sqrt{N}$, where $N$ is the number of training samples. We used the rectangular shape for the map and the hexagonal lattice as a shape for the neurons. Each neuron has six neighbors relying on a Gaussian neighborhood function. In our process, we targeted the agglomerations of the map which are dense and containing an important number of words.

3.4.1 Tracking the cardinality

At the end of the training step, the similarity between the inputs and all the neurons of the map is computed. The neuron that is the closest is called the best-matching unit (BMU). We computed the hit histogram for the map as follows: The BMU of each data sample is calculated, and a counter increases in the target neuron each time it is the BMU of a word. Neurons with higher values are considered interesting neurons from which we should extract the tags. This was achieved using the Pareto selection algorithm to achieve this.

3.4.2 Tracking the density

We excluded isolated neurons. We mean neurons whose distances from adjacent neurons are important. We relied on the assumption that effective tags are located on neurons on the map where we should have important density of words and neighboring neurons are close (in terms of distance). For this reason, we computed the U-matrix [35] from the obtained map. The U-matrix allows showing similarity between the neurons. The U-matrix is obtained by calculating the average of distances between each neuron and its neighbors. After
that, from the U-matrix we computed the lower quartile. All neurons whose distances from adjacent neurons are greater than the lower quartile were eliminated. The retained tags are the words that are mapped to retained neurons.

Algorithm 1: Exploring the SOM and extracting tags

Result: tags

SN = Set of neurons of SMap;
HH = Compute hit histogram from SMap;
Eliminate from SN all neurons having a score less than two;
Order SN according to their neurons scores;
Eliminate from SN neurons by applying the Pareto principle;
UM = Compute U-Matrix;
LQ = Compute the lower Quartile;
foreach (Neuron in SN) do
  if (Distance between neuron and each neighborhood neuron > LQ) then
    Eliminate neuron from SN
  end
end

tags = words mapped in the neurons of SN.

3.5 Complexity of our approach

To evaluate the time of our approach, we calculate the complexity of the SOM clustering step. The other steps such as comments scrapping, feature computing and mining are not time-consuming. The computational cost for SOM exhibits a linear complexity. The processing time is proportional to the number of candidates words n. Therefore, the complexity is on the order of O(n). Thus, the complexity of our approach is O(n).

4 Experiments

4.1 YouTube dataset

YouTube 8M [36] is one of the Google’s latest contributions to research in machine learning and artificial intelligence. Having started in 2017, it is a dataset with 8 million video IDs labeled with more than 4000 classes. In 2018, the competing data have been gathered into a dataset that contains 6.1 million videos and the number of classes has been restricted to 3862 only. Videos are selected according to some criteria such as being public and having at least 1000 views, lasting between 120 and 500 seconds and having associations with at least one entity of the target vocabulary (one of the 3862 words mentioned above). Each video is labeled by one of the 3682 words by inspecting its internal frame structure. The videos of this dataset belong to different categories: Arts & Entertainment, Autos & Vehicles, Beauty & Fitness, Books & Literature, Business & Industrial, Computers & Electronics, Finance, Food, Drink, Games, Health, Hobbies & Leisure, Home & Garden, Internet & Telecom, Jobs & Education, Law & Government, News, People & Society, Pets & Animals, Real Estate, Reference, Science, Shopping, Sports & Travel [37,38]. From this dataset, we selected 1000 videos belonging to different categories.

4.2 Evaluation rates

The main objective of the experiments was to test the quality of the generated tags according to a ground truth. In the literature, we could not find labeled dataset where we have the video the comments and the ground-truth tags. The only solution that we found is to compare the generated tags according to tags made by users (ground truth) and to the title. The title summarizes the video in a few words. Our goal is to test whether our approach is able to catch the most important concepts of the original video. Moreover, YouTube 8M comes with

- The ground-truth tags submitted with the video by its owner,
- Words constituting the title of the video,
- Labels added to videos by Google.

We evaluated our approach by computing the recall, precision, and F1 score between the tags extracted by our approach and the tags of the ground truth. These are computed as follows:

\[
\text{Recall} = \frac{\text{#tags correctly detected}}{\text{# ground-truth tags}} \quad (10)
\]
\[
\text{Precision} = \frac{\text{#tags correctly detected}}{\text{# total detected tags}} \quad (11)
\]
\[
F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (12)
\]

4.3 Evaluation against video titles

The title of a video is a piece of metadata that YouTube highlights greatly. We computed only the recall rates. Indeed, the precision rate is not really significant in this case. We may find correctly extracted tags, but not included in the title words. “Ground-truth tags” are words that make up the title. The obtained results are displayed in Table 2. It can be noticed that the extracted tags in many videos match a lot with the titles. Regardless of the categories and the used distance (Google distance and WordNet-based distance), the results are similar. It essentially depends on the length of the title. The shorter the title is, the better the recall rate will be. This evaluation gives us important perspectives when dealing with
Table 2 Evaluation against videos titles

| YouTube categories          | Google distance | WordNet distance |
|----------------------------|-----------------|------------------|
|                            | Recall (%)      | Recall (%)       |
| Arts & Entertainment        | 68.8            | 65.2             |
| Autos & Vehicles           | 76.4            | 78.5             |
| Beauty & Fitness           | 53.6            | 59.4             |
| Books & Literature         | 56.2            | 62.8             |
| Business & Industrial      | 70.4            | 68.4             |
| Computers & Electronics    | 82.5            | 75.6             |
| Finance                    | 65.0            | 58.1             |
| Food & Drink               | 68.4            | 73.5             |
| Games                      | 75.5            | 78.6             |
| Health                     | 79.5            | 85.6             |
| Hobbies & Leisure          | 54.8            | 59.5             |
| Home & Garden              | 65.7            | 62.2             |
| Internet & Telecom         | 76.8            | 79.5             |
| Jobs & Education           | 58.4            | 62.2             |
| Law & Government           | 60.4            | 58.7             |
| News                       | 69.8            | 67.5             |
| People & Society           | 58.7            | 51.5             |
| Pets & Animals             | 63.5            | 68.7             |
| Real Estate                | 69.8            | 72.2             |
| Reference                  | 65.5            | 72.5             |
| Science                    | 73.5            | 75.6             |
| Shopping                   | 68.4            | 70.5             |
| Sports                     | 72.6            | 76.5             |
| Travel                     | 75.5            | 73.7             |

untitled videos in social media. We can rely on the comments to suggest titles for this kind of videos.

4.4 Evaluation against uploaders tags

Table 3 displays the performance of the proposed approach to extract tags according to the uploaders’ suggestions. Uploaders’ tags are considered as ground-truth tags. As a first observation, we can notice that the results are not the same for all categories. For categories such as sports, science and technology, travel, and gaming, the results are clearly better. This is not surprising because according to YouTube Academy [39] these categories are the most popular. The most popular means the most viewed and hence they are well commented.

Moreover, for categories such as “Books & Literature” or “Beauty & Fitness” or “Arts & Entertainment,” the internet users generally tend to make reviews rather than describe the content of the video.

However, in such categories as “Computers & Electronics” or “Auto & Vehicles,” people make reviews and describe the content at the same time. Typical examples of products unboxing videos can be suggested here. (They mention the name of the product, the brand, the options, etc.)

In addition, regarding the precision rates, we can notice the ability of the proposed approach to extract correct tags from comments. This proves that we can rely on this approach to make tags enrichment. This also confirms that self-organizing maps are able to detect tags clusters efficiently.

The F1 scores confirm the good performance of our approach. The recall and the precision rates are close, and this is why the values of F1 scores are important. This shows that our approach is both precise and exact.

When comparing the results obtained via NGD with those obtained with Leacock distances, we can confirm that NGD is more efficient. Indeed, a quick glance at the uploaders’ tags (the ground truth) allows us to notice that they are not always linguistically related; however, contextually they are. The contextual relationship could not be quantified through Leacock distances. They depend on the contexts. For example, the link between coronavirus and Wuhan is contextual and not linguistic. For this reason, tags generated via NGD are more liable than WordNet linguistic Leacock similarity distance.

4.5 Evaluation against YouTube 8M labels

The YouTube 8M consists of a set of videos labeled with entities. Labels include both coarse and fine-grained entities, which have been semi-automatically curated and manually verified by three raters to be visually recognizable. The ground-truth video labels are the main themes of each video, as determined by a YouTube video annotation system using content, metadata, contextual, and user signals. The number of ground-truth labels per video varies from 1 to 23, with an average of 3.01 per video. The 60th and 80th percentiles of labels/video are 3.0 and 4.0, respectively. Hence, it can be noticed that the number of labels per video is not so important. Labels of the dataset summarize just the general context of the videos. These labels do not evoke all the video concepts, but only those that summarize the general context. We could not consider them as exhaustive tags, but it is still interesting to see whether the proposed approach is able to detect them. For this reason, we computed only the recall rate. Table 4 shows the performance of the proposed approach to extract these labels. The achieved results are generally interesting. First, they showed that the revealed comments tags can summarize the general context of the videos. Second, they proved that the proposed approach is able to detect important concepts evoked by the videos. Finally, the obtained results using Google distance and WordNet-based distance are close. This is due to the fact that the number of labels for each video is small, with an average of 3.01 per video. Overall, it can be confirmed that using NGD or Leacock distances, the results are interesting.
### Table 3  Recall and precision computed according to tags creator

| YouTube categories       | Google distance | WordNet distance |
|--------------------------|-----------------|------------------|
|                          | Recall (%)      | Precision (%)    | F1 (%) | Recall (%)      | Precision (%) | F1 (%) |
| Arts & Entertainment     | 72.5            | 72.3             | 72.4   | 63.5            | 75.0           | 68.77   |
| Autos & Vehicles        | 93.0            | 92.0             | 92.5   | 86.4            | 96.0           | 90.5    |
| Beauty & Fitness        | 77.0            | 81.0             | 78.5   | 81.0            | 82.5           | 81.75   |
| Books & Literature      | 65.0            | 63.0             | 64.0   | 56.9            | 66.8           | 61.45   |
| Business & Industrial   | 88.0            | 74.0             | 80.4   | 72.0            | 78.5           | 75.1    |
| Computers & Electronics | 90.0            | 87.0             | 88.45  | 87.5            | 89.5           | 88.5    |
| Finance                 | 89.0            | 82.0             | 85.35  | 87.1            | 80.6           | 83.72   |
| Food & Drink            | 72.0            | 61.0             | 66.0   | 72.6            | 68.2           | 70.33   |
| Games                   | 86.0            | 79.0             | 82.35  | 82.6            | 85.8           | 84.16   |
| Health                  | 93.0            | 92.0             | 92.5   | 88.2            | 90.4           | 89.3    |
| Hobbies & Leisure       | 75.0            | 56.0             | 64.12  | 65.8            | 49.6           | 56.56   |
| Home & Garden           | 75.0            | 71.0             | 72.94  | 69.8            | 66.2           | 67.95   |
| Internet & Telecom      | 92.0            | 84.0             | 87.81  | 87.4            | 88.5           | 87.94   |
| Jobs & Education        | 76.0            | 74.0             | 75.0   | 70.5            | 75.2           | 72.77   |
| Law & Government        | 82.0            | 78.0             | 79.95  | 71.0            | 82.1           | 76.2    |
| News                    | 94.0            | 89.0             | 91.43  | 88.4            | 85.2           | 86.77   |
| People & Society        | 76.0            | 68.0             | 71.77  | 81.2            | 70.3           | 75.35   |
| Pets & Animals          | 92.0            | 83.0             | 87.26  | 88.2            | 90.5           | 89.33   |
| Real estate             | 81.0            | 77.0             | 78.94  | 76.3            | 68.3           | 72.0    |
| Reference               | 72.0            | 67.0             | 69.41  | 65.2            | 68.0           | 66.57   |
| Science                 | 92.0            | 95.0             | 93.47  | 87.3            | 88.3           | 87.8    |
| Shopping                | 83.0            | 82.0             | 82.5   | 78.6            | 80.6           | 79.6    |
| Sports                  | 95.0            | 90.0             | 92.43  | 88.4            | 82.5           | 85.34   |
| Travel                  | 88.0            | 88.0             | 88.0   | 82.2            | 90.4           | 86.1    |

### Table 4  Evaluation against YouTube 8M labels titles

| YouTube categories       | Google distance | WordNet distance |
|--------------------------|-----------------|------------------|
|                          | Recall (%)      | Recall (%)       |
| Arts & Entertainment     | 78.8            | 75.2             |
| Autos & Vehicles        | 92.1            | 89.2             |
| Beauty & Fitness        | 70.4            | 66.2             |
| Books & Literature      | 76.2            | 72.8             |
| Business & Industrial   | 84.5            | 85.3             |
| Computers & Electronics | 85.7            | 82.3             |
| Finance                 | 74.6            | 70.3             |
| Food & Drink            | 80.7            | 78.4             |
| Games                   | 89.2            | 82.2             |
| Health                  | 88.2            | 83.2             |
| Hobbies & Leisure       | 68.1            | 63.2             |
| Home & Garden           | 80.2            | 75.3             |
| Internet & Telecom      | 91.3            | 90.7             |
| Jobs & Education        | 73.2            | 70.2             |
| Law & Government        | 71.2            | 65.9             |
| News                    | 88.7            | 81.5             |

### Table 4 continued

| YouTube categories       | Google distance | WordNet distance |
|--------------------------|-----------------|------------------|
|                          | Recall (%)      | Recall (%)       |
| People & Society         | 67.9            | 65.3             |
| Pets & Animals           | 84.2            | 81.7             |
| Real Estate              | 75.7            | 74.3             |
| Reference                | 79.4            | 80.2             |
| Science                  | 90.5            | 88.9             |
| Shopping                 | 83.5            | 79.2             |
| Sports                   | 85.4            | 82.9             |
| Travel                   | 87.2            | 86.3             |

### Table 5  Execution time (seconds)

| Phase                   | Google distance | WordNet distance |
|-------------------------|-----------------|------------------|
| Training                | 8.67            | 9.2              |
| Mining                  | 2.4             | 2.59             |
### Table 6  Comparing recall and precision rates of our work and of YAKE computed according to tags creator

| YouTube categories          | Our work | YAKE |
|----------------------------|----------|------|
|                            | Recall (%) | Precision (%) | F1 (%) | Recall (%) | Precision (%) | F1 (%) |
| Arts & Entertainment       | 72.5      | 72.3  | 72.4  | 75.5      | 51.2      | 61    |
| Autos & Vehicles           | 93.0      | 92.0  | 92.5  | 90.6      | 67.5      | 77.36 |
| Beauty & Fitness           | 77.0      | 81.0  | 78.5  | 75.0      | 69.4      | 72.1  |
| Books & Literature         | 65.0      | 63.0  | 64.0  | 62.3      | 49.2      | 55    |
| Business & Industrial      | 88.0      | 74.0  | 80.4  | 82.0      | 54.2      | 65.26 |
| Computers & Electronics    | 90.0      | 87.0  | 88.45 | 89.2      | 71.1      | 80.66 |
| Finance                    | 89.0      | 82.0  | 85.35 | 86.0      | 59.0      | 70.0  |
| Food & Drink               | 72.0      | 61.0  | 66.0  | 71.0      | 58.1      | 64.0  |
| Games                      | 86.0      | 79.0  | 82.35 | 80.5      | 69.1      | 74.36 |
| Health                     | 93.0      | 92.0  | 92.5  | 90.4      | 80.2      | 85.0  |
| Hobbies & Leisure          | 75.0      | 56.0  | 64.12 | 73.4      | 41.0      | 52.61 |
| Home & Garden              | 75.0      | 71.0  | 72.94 | 71.5      | 54.3      | 61.72 |
| Internet & Telecom         | 92.0      | 84.0  | 87.81 | 93.5      | 65.1      | 76.75 |
| Jobs & Education           | 76.0      | 74.0  | 75.0  | 78.1      | 55.0      | 64.54 |
| Law & Government           | 82.0      | 78.0  | 79.95 | 79.0      | 62.2      | 69.6  |
| News                       | 94.0      | 89.0  | 91.43 | 95.0      | 59.2      | 73    |
| People & Society           | 76.0      | 68.0  | 71.77 | 74.2      | 45.3      | 56.25 |
| Pets & Animals             | 92.0      | 83.0  | 87.26 | 90.6      | 61.1      | 73    |
| Real estate                | 81.0      | 77.0  | 78.94 | 79.1      | 51.0      | 62.0  |
| Reference                  | 72.0      | 67.0  | 69.41 | 75.0      | 52.7      | 62.0  |
| Science                    | 92.0      | 95.0  | 93.47 | 90.4      | 68.2      | 67.74 |
| Shopping                   | 83.0      | 82.0  | 82.5  | 80.4      | 58.0      | 67.38 |
| Sports                     | 95.0      | 90.0  | 92.43 | 98.2      | 60.5      | 74.87 |
| Travel                     | 88.0      | 88.0  | 88.0  | 84.0      | 70.0      | 76.36 |

### 4.6 Execution time

As mentioned previously, the SOM input contains candidate words retained after the filtering step. Their number does not exceed a few hundred in general. Scalability is an important criterion. We measured the execution time of the two phases: SOM clustering and mining. The results are presented in Table 5. They show that the clustering is the time-consuming phase regardless of the used distance. The tests were performed on a Core I7 2.33 GHz CPU and 8 GB RAM computer. The average time does not exceed 10 seconds, which is acceptable. If we use a more efficient machine, we could significantly reduce the execution time.

### 4.7 Comparison results

We compared our work to YAKE which is a recent work proposed by Campos et al. [17]. The APIs for this work are published for free on GitHub. YAKE is an unsupervised automatic keyword extraction method that relies on statistical text features extracted from single documents to select the most relevant tags from a text. We have extracted the comments from the YouTube video, and we presented them to YAKE. The results of the comparison are shown in Table 6. We obtained comparable recall rates. Like our work, YAKE was able to detect the most important tags. This confirms that the use of unsupervised techniques is adequate to extract the tags. However, the main problem lies in the precision rates. YAKE has extracted many false tags, and this is because it is a general tag extraction work. To extract tags from comments, we need to consider two important criteria, redundancy, and closeness. The semantic closeness of tags is not considered by YAKE.

### 5 Conclusion

This paper introduced an approach for extracting tags from comments posted on social media videos. The novel system exploits collective knowledge embedded in user-generated comments. It is based on the observation that comments can contain information describing the content of the video. We also relied on natural language processing techniques, as
well as statistical techniques and pattern recognition ones, to retrieve eventual tags.

It was proven that projecting the user’s comments on a self-organizing maps allows the extraction of effective tags. We do not pretend that the comments can be a perfect solution for tagging videos. We just demonstrated that comments could help to tag videos. They could be used to enrich existing uploaders’ tags, or just validate or refine them.

The proposed approach was tested on YouTube 8M dataset. The obtained results are encouraging and show that when there are good quality comments, good quality tags will follow, especially for some categories of videos such as sports, science and technology, travel, and gaming.

For this reason, we are planning to propose an approach that evaluates the quality of comments and their ability to provide effective tags in a future study.

Acknowledgements The author would like to acknowledge the technical support of the Faculty of Economics and Management of Sfax and the Tunisian Ministry of Higher Education and Scientific Research.

Author Contributions All the contributions proposed in this paper and mentioned in Sect. 2.3 are made by the ME.

Funding This work was done due to the financial support from the General Direction of Scientific Research and Technological Renovation (DGRSRT), Tunisia, under the PEJC program.

Availability of data and material The data that support the findings of this study are available from the corresponding author upon reasonable request.

Declarations

Conflict of interest The author declares he has no conflict of interest.

References

1. Mazloom M, Li X, Snoek CGM (2016) 5-TagBook: a semantic video representation without supervision for event detection. IEEE Trans Multimedia Arch 18(7):1378–1388
2. Siersdorfer S, San Pedro J, Sanderson M (2009) Automatic video tagging using content redundancy. In: Proceedings of ACM SIGIR, pp 395–402
3. Zhao W, Wu X, Ngo CW (2010) On the annotation of web videos by efficient near-duplicate search. IEEE Trans Multimedia 2:448–461
4. Viana P, Pinto JP (2017) A collaborative approach for semantic time-based video annotation using gamification, Human Centric Comput Inf Sci 7(1)
5. Ballan L, Bertini M, Uricchio T, Del Bimbo A (2015) Data-driven approaches for social image and video tagging. Multimedia Tools and Applications 74(4):1443–1468
6. Ballan L, Bertini M, Serra G, Del Bimbo A (2015) A data-driven approach for tag refinement and localization in web videos. Comput Vis Image Underst 140:58–67
7. Baraldi L, Grana C, Cucchiara R (2017) A video library system using scene detection and automatic tagging. In: Italian research conference on digital libraries proceedings, pp 56–67
8. Zhu X, Huang Z, Cui J, Shen HT (2013) Video-to-shot tag propagation by graph sparse group lasso. IEEE Trans Multimedia 15:633–646
9. Wang M, Hong R, Li G, Zha Z-J, Yan S, Chua T-S (2012) Event driven web video summarization by tag localization and key-shot identification. IEEE Trans Multimedia 14:975–985
10. Oorschot G, ERP M, Dijkshoorn C (2012) Automatic extraction of soccer game events from twitter, workshop on detection, representation, and exploitation of events in the semantic
11. Tang A, Boring S (2012) EpicPlay: crowd-sourcing sports video highlights. In: SIGCHI conference on human factors in computing systems. ACM, pp 1569–1572
12. Chiraratanasopha B (2021) Sentimental analysis and keyword extraction from Thai users of facebook in COVID-19 period, progress in applied. Sci Technol 11:66–72
13. Chou CL, Tsai MH, Chao CH et al (2014) Automatic restaurant information and keyword extraction by mining blog data for Chinese restaurant search. In: Proceedings of Pacific-Asia conference on knowledge discovery and data mining. Taiwan, pp 700–711
14. Yang W, Yang Z (2016) Keyword extraction method over blog community. In: Proceedings of the international conference on fuzzy systems and knowledge discovery, pp 1553–1556
15. Marujo L, Ling W, Trancoso I, Dyer C, Black AW, Gershman A et al (2015) Automatic keyword extraction on twitter. In: Proceedings of the annual meeting of the association for computational linguistics and the international joint conference on natural language processing, pp 637–643
16. Park J, Kim J, Lee J-H (2014) Keyword extraction for blogs based on content richness. J Inf Sci 40(1):38–49
17. Campos R, Mangaravite V, Pasquali A, Jorge A, Nunes C, Jatowt A (2020) YAKE! keyword extraction from single documents using multiple. J Inf Sci 509:257–289
18. Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biol Cybern 43:59–69
19. Youtube API—Youtube Data API.https://developers.google.com/youtube/v3/docs/comments/list
20. Bird S, Klein E, Loper E (2009) Natural language processing with python. O’Reilly Media Inc
21. Anthony L (2019) Computer software. University of Waseda, Tokyo, Japan. http://www.laurenceanthony.net/
22. Salton G, McGill MJ (1986) Introduction to modern information retrieval. McGraw-Hill
23. Talabis M, McPherson R, Miyamoto I, Martin J (2015) Information security analytics: finding security insights, patterns, and anomalies in big data, pp 123–150
24. Pacella M, Griecco A, Blaco M (2016) On the use of self-organizing map for text clustering in engineering change process analysis: a case study. In: Computational intelligence and neuroscience, pp 1–11
25. Klami M, Lagus K (2006) Unsupervised word categorization using self-organizing maps and automatically extracted morphs. In: IDEAL, intelligent data engineering and automated learning, pp 912–919
26. Ellouze M, Karray H, Soltana WB, Alimi AM (2007) Utilisation d’un journal télévisé. In: Proceedings of international conference of the annual meeting of the association for computational linguistics and the international joint conference on natural language processing, pp 1553–1556
27. Elouzoue M, Karray H, Soltana WB, Alimi AM (2007) Utilisation d’un journal télévisé. In: Proceedings of international conference of the annual meeting of the association for computational linguistics and the international joint conference on natural language processing, pp 1553–1556
28. Ayadi T, Ellouze M, Hamdani TM, Alimi AM (2013) Movie scenes extraction from big data, pp 123–150
29. Bollegala D, Yutaka M, Mitsuru I (2007) Measuring semantic similarity between words using web search engines. In: Proceedings of international world wide web conference, pp 757–766
30. Chen PI, Lin SJ (2010) Automatic keyword prediction using Google similarity distance. Exp Syst Appl 37(3):1928–1938
31. Xian Z, Weber K, Fesenmaier DR (2008) Representation of the online tourism domain in search engines. J Travel Res 47(2):137–150
32. Cilibrasi R, Vitanyi P (2007) The google similarity distance. IEEE Trans Knowl Data Eng Arch 19(3):370–383
33. Fellbaum C (1998) WordNet: an electronic lexical database, language, speech, and communication. MIT Press, Cambridge, USA
34. Vesanto J, Himberg J, Alhoniemi E, Parhankangas J (2005) SOM toolbox for Matlab 5. http://www.cis.hut.fi/somtoolbox/
35. Ultsch A, Siemon HP (1990) Kohonen’s self-organizing feature maps for exploratory data analysis. In: Proceedings of the international neural network conference. Kluwer Academic Press, Dordrecht
36. YouTube 8M—Large-scale labeled video dataset. https://research.google.com/youtube8m/
37. Abu-El-Haija S, Kothari N, Lee J, Natsev P, Toderici G, Varadarajan B, Vijayanarasimhan S (2016) Youtube-8m: a large-scale video classification benchmark. arXiv
38. Google I/O 2013—semantic video annotations in the Youtube Topics API: theory and applications. https://www.youtube.com/watch?v=wf_77zH-vQ
39. YouTube Academy—last visited, April 2020. https://creatoracademy.youtube.com/page/home?hl=fr

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.