A Textual Filtering of HOG-based Hierarchical Clustering of Lifelog Data

Mihai Dogariu, Bogdan Ionescu

Multimedia Lab @ CAMPUS, Politehnica University of Bucharest, Romania
mdogariu@imag.pub.ro,bionescu@alpha.imag.pub.ro

Abstract. In this paper we address the issue of life logging information retrieval and we introduce an approach that uses the output of a hierarchical clustering of data via assessing word similarities. Word similarity is computed using WordNet and Retina ontologies. We have tested our method during the 2017 ImageCLEF Lifelog challenge, the Summarization subtask. We discuss the performance, limitations and future improvements of our method.

Keywords: Lifelog, WordNet, word similarity, hierarchical clustering, concept detector, Retina

1 Introduction

Lifelogging is the process of tracking your own activities at every moment of the day. But this is not limited only to actions as it also addresses the places that you visit and the people that you engage into activities with. A formal definition describes lifelogging as *"a form of pervasive computing, consisting of a unified digital record of the totality of an individuals experiences, captured multimodally through digital sensors and stored permanently as a personal multimedia archive"* [1].

Lifelogging consists of best describing all events within a time frame from ones life. This is usually achieved by wearing a camera which takes automatic pictures at certain time intervals during the entire day, thus capturing a great amount of information from the wearer's point of view. However, this does not offer temporal or spatial information other than what it can be extracted from the details of the captured images. In order to solve this, certain devices can be used to add timestamps and geolocation to the images. In addition, there also exists devices that offer information about numerous other aspects concerning the lifelogger's status and the surrounding environment such as pulse, temperature, pH or acoustic sensors. As it can be seen, only the environment limits the type of data that can be collected and all these details build up leading to an increasingly accurate description of the lifelog.

Extensive details about lifelogging can also be found in [2, 3]. In these 2 papers, different techniques of acquiring data are presented and they offer important insights on the problems that arise with this new trend. Recent technological advances have aided the growing trend of lifelogging by offering longer battery life for the wearable gadgets, a greater variety of sensors that augment the observable aspects of all events and also higher accuracy for the provided data. However, all these aspects have the drawback of requiring complex processing steps. With the technical problems related to acquisition or storage of information being solved by the state of the art devices there have been obtained large volumes of data which need to be interpreted.

As it would be a tedious process to manually annotate and interpret all the data from a lifelog, an increasing interest in developing techniques that perform this automatically has emerged. Therefore, Information Retrieval systems fit the requirements for this job, potentially solving the problems posed by the vast quantity of data. A rigorous comparative benchmarking campaign regarding Lifelogging was conducted by the NTCIR initiative [4] and has now also been adapted by the CLEF initiative in the ImageCLEF Lifelog task [5].

This paper describes our participation to ImageCLEF Lifelog Summarization Task, one of the 4 tasks approached by the ImageCLEF benchmarking campaign [6]. The summarization task aims at clustering a lifelog database given 10 topics. Each of these topics consists of a short paragraph which briefly describes what is considered relevant and what is not for that specific task. Our work focused on finding a correlation between the textual description of a topic and the confidences provided by the concept detector for each image in the given database. For each image we computed the Wu-Palmer similarity measure [7] between the most relevant concepts and a selection of words from the topic description. What "relevant concepts" means and how the selection has been done is described in the following Section. Each similarity measure is then weighted by the confidence assigned by a concept detector for the relevant concept, thus obtaining a score for each image. The images from the database are clustered with an off-the-shelf hierarchical clustering implementation. In the end, the similarity score is used to sort and select the best candidates from these clusters. Related works have been described in [4] with [8] and [9] being closest to our approach as they also computed a similar WordNet [10] based similarity distance in their algorithms.

The remainder of the paper is organized as follows. In Section 2 we explain our proposed system, in Section 3 we present our experimental results, followed by the conclusions in Section 4.

2 The proposed approach

Due to the fact that the given data contained a large variety of information, our algorithm tries to capture this aspect by approaching several tasks in parallel. It also mixes up textual and visual information in the process of selecting the best candidates for the requirements of the task. In Fig. 1 a general diagram of the involved processes shows that visual and textual descriptions interleave. An important aspect of this algorithm is that it relied solely on the information provided by the organizers and no additional annotations or external data have been used.

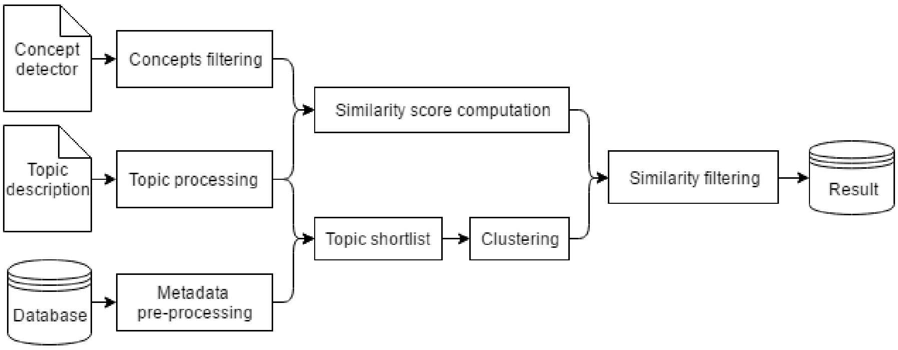

Fig. 1. General diagram of the proposed algorithm.

The algorithm starts by analyzing the output of the concept detector provided by the organizers and selecting for each image the most probable concepts only. Each topic out of the list of 10 is then parsed such that relevant words are kept only. Also, information regarding location, activity and the targeted user are extracted. The image database comes with an .xml file that describes each image in terms of activity, location, timestamp and descriptive information related to the user the image belongs to. The similarity score is computed using WordNet's builtin similarity distance functions and this concludes the fundamental part of the textual part. The images undergo an elimination step imposed by the topic restrictions and thus the number of items of interest is greatly reduced. This shortlist of images is then subject to a clustering step and, finally, the results are pruned with the help of the similarity scores presented before.

2.1 Image concepts

An important input to our system is represented by the list of concepts along with the respective confidences for each image from the database. This has been obtained by passing each image through the CAFFE CNN-based visual concept detector and storing the results, as described by the organizing team [5]. This concept detector acts as a soft classifier with 1000 classes and it outputs a given confidence level to each class. It is very debatable if this concept detector is fit for such a task since it covers a large and diverse range of classes, many of which are not related to the objects depicted or seen on a daily basis by the lifeloggers from the database. It is worth mentioning that the vast number of concepts may affect their accuracy and, consequently, the results of our system.

It is obvious that not all 1000 concepts are relevant for one image as it is difficult even for humans to detect more than a few tens of concepts in a real-life picture as are the ones from the database. Therefore, it was mandatory to filter out the unnecessary concepts and we chose to do so based on the confidence score that was provided by the concept detector. Thus, it became a matter of design to select a certain threshold above which a concept would be qualified as relevant. We chose a statistic approach and set the threshold dependent on the confidence distribution of each individual image.

Let $C^{(i)} = (c_0, c_1, ..., c_{999})$ be the set of confidences for any image i in the dataset. By imposing a threshold:

$$t = \mu + 3 \times \sigma, \tag{1}$$

with μ being the mean and σ being the standard deviation over $C^{(i)}$. The set of relevant concepts becomes:

$$C^{(i)}_{relevant} = \{c_i \mid \forall c_i \in C^{(i)} \text{ s.t. } c_i > t\}, \tag{2}$$

thus giving us some insightful information about the concepts that were detected in the images.

The first notable aspect is that the number of relevant concepts whose confidence is greater than t differs from one image to another due to the standard deviation. Since all confidences add up to 1 for any image it is clear that the mean will always be 0.001, leaving the standard deviation as the only variable parameter when computing the threshold. As for most distributions, there is a very narrow set, if any, of concepts that have a confidence higher than the given threshold due to its high value. The results can be interpreted as follows: if there are some concepts with a very high confidence score they will raise the standard deviation of the entire set and will eliminate other less confident concepts; if there are no such highly confident concepts then the algorithm will select the most probable few from the set, leading to a larger set of concepts. All in all, this exploits highly confident concepts and it also takes into account the cases where there is no such certainty and where several guesses are bound together to form a less probable description of an image.

Since each concept is described as one or several words we used WordNet to find the common synset to all the words that describe a concept and use it to describe the concept with one word only. This was also a preliminary step for computing the similarity score described in Section 2.5.

2.2 Topic analysis

The topics given by the organizers represent the search query for our information retrieval system. There are a total of 10 topics, each being described by a title, a short summary and a slightly longer detailed description. Most of the times, the information from the title and the summary is also present in the detailed description so we can ignore the words from the title and the summary. Moreover, the detailed descriptions followed a general pattern, stating which user was targeted by that specific query, what action he was involved in and the place where he was at the moment when the picture was taken.

The topics also impose restrictions on the conditions under which an image is relevant or not with some of them being very strict on this matter. However, all images that are blurry, out of focus or where the hands of the lifelogger covered most of the picture are considered irrelevant.

As the topic descriptions consisted of a text file, the analysis of the topics falls under the NLP processing, covered by WordNet's embedded tools. In other words, for each of the 10 topics we did as follows: we stripped the general restrictions part as it appeared for each topic and did not help discriminate between them and extracted only the keywords from the text formed by joining the title, summary and description of a topic. Another step was to remove all stopwords, which offer little to no information in a sentence. As WordNet was designed for nouns and verbs only, we kept these parts of speech from the remaining list of words. We are aware that some important information is lost when removing adverbs, adjectives and negations but as NLP is not our main research field we adopted this simplified parsing algorithm.

In the end, for each topic we obtain some coarse, but fundamental information related to that specific query. As an example, the next topic:

> *T1. In a Meeting 2*
> *Query: Summarize the activities of user u1 in a meeting at work.*
> *Description: To be considered relevant, the moment must occur at meeting room and must contain at least two colleagues sitting around a table at the meeting. Meetings that occur outside of the work place are not relevant. Different meetings have to be summarized as different activities. Blurred or out of focus images are not relevant. Images that are covered (mostly by the lifelogger's arm) are not relevant.*

is now summarized as the set of words $T^{(1)}$ ={*'activities', 'u1', 'meeting', 'work', 'occur', 'room', 'contain', 'colleagues', 'sitting', 'table', 'meetings', 'place'*}. There is an obvious shortening of the description, but the meaning of the new context is still understandable to any reader. Automatically extracting the keywords from a sentence is still an open problem on which the NLP community is still working.

2.3 Narrowing the list of images

Metadata pre-processing involved parsing the xml file associated with the image database and transferring the useful information in a more user-friendly format such as a matrix where we stored each important attribute on a different column for each image. Among these attributes we note the user id, image id and path, the activity and the location where the said picture was taken. It is worth mentioning that the location and activity tags are not very specific, thus offering a wide range for extrapolating on them only. We discovered 6 different activities (bus, car, cycling, running, transport, walking) and over 100 semantic locations as they are tagged in the Moves App [11], but we also note that images are not mandatory tagged with the activity or location. Some of them do not posses this attribute and we have filled their corresponding attribute value with 0.

The provided database consisted of images belonging to 3 lifeloggers or users as it is described in the lifelog task overview [5] and each topic addressed one particular user only. This meant that the other 2 users could be ignored in the decision process. The same logic applies to the activity and location fields, where the topic description would eliminate certain candidate images because of their respective metadata, e.g. if the query asks for images where the user u1 is in a meeting at work it is not possible to select images where the location indicates something else than user u1's workplace (or 0 if the workplace has not been annotated in the Moves App) or where the activity is bus, car, cycling, running or transport. This type of content interpretation allowed us to create a shortlist of images for each distinct topic which we then used in the clustering part.

2.4 Image clustering

For the clustering part we implemented a hierarchical clustering algorithm based on the Histogram of Oriented Gradients (HOG) [12] extracted from each image in the previously described shortlist. For each image we cropped the edges as follows: 100 pixels from the top edge, 128 from both left and right edge and 25 pixels from the bottom edge. We chose these values as the users wear their camera around their neck and, quite often, the camera is slightly tilted upwards, especially when the user is sitting down, thus adding some unnecessary information in the upper part of the image or the user's clothing covers one of the other 3 edges.

The cropped images underwent a resize step as well, bringing them to a format of 64×128 pixels, so that the HOG extraction process could be performed. We have used a simple builtin HOG extractor from Python's cv2 module and stopped the hierarchical clustering algorithm when 30 clusters were formed.

2.5 Similarity score

The process of finding semantic links between any two words is a very complex task for NLP and is still an open problem. However, WordNet offers a large lexical database for English which incorporates not only a glossary, but also a tree-like structure that connects semantic meanings starting from a root node, which is the most abstract concept, and descends to more and more particular concepts. Even so, this tree-like structure has been developed for nouns and verbs only, this being the motive behind the algorithm used for topic description summarization.

WordNet has its information organized in sets of cognitive synonyms (synsets) interleaved by means of conceptual-semantic and lexical relations [10]. This means that the number of edges that have to be passed in order to reach from one synset to another gives information about how similar these two synsets are. Having this in mind, a first approach to finding a similarity measure would be to follow the shortest path between two nodes, count the number of edges along the way and then set the similarity measure to be inversely proportional to the previously found number. A maximum similarity score of 1 would be obtained when the two synsets are the same. However, this does not take into account the depth of the node from which the two synsets descend. This node is known as the Least Common Superconcept (LCS) and the distance from the root node to this node gives a measure of abstractness.

One word similarity measure that takes into account both the path length between two concepts and the depth of their LCS is the Wu-Palmer similarity measure. Given the tree structure from Fig. 2 the Wu-Palmer similarity distance between the concepts C_1 and C_2 is

$$d_{wup}(C_1, C_2) = \frac{2 \times N_3}{N_1 + N_2 + 2 \times N_3}, \tag{3}$$

where C_3 is the LCS, N_3 is the depth of the LCS from the root node and N_1 and N_2 are the depths of C_1 and $C2$, respectively, from the LCS.

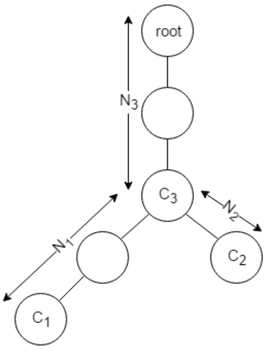

Fig. 2. Wu-Palmer similarity distances on a tree.

This approach gives higher similarity values to the pairs of concepts which have the LCS deeper in the tree structure and the score will always have a value in the range $]0, 1]$. The value cannot be 0 because the depth of the LCS can never be 0 (the root of such a taxonomy has a depth of 1), and it will be 1 only when the two synsets coincide.

Another problem that arises when comparing the similarity of two words is that the previously described method compares the distance between synsets (or concepts) and not words.A word can, and most of the time will, belong to more than one synset as we often encounter words with more than one meaning. A simple way to imagine this is to think of a thesaurus and when we search the meaning of a word we get numerous entries. Each of those entries represents a synset. Therefore, when we want to compare two words from a similarity point of view we actually need to first select the most appropriate synsets for each of them and compare those two synsets. Since it was not feasible to manually find the best matching pairs of synsets corresponding to a pair of words we picked the one that ranked the highest only. Therefore, the Wu-Palmer distance computed between two words w_1 and w_2 becomes

$$d(w_1, w_2) = max(d_{wup}(s_i, s_j)), \forall i, j \text{ s.t. } s_i \in S(w_1), s_j \in S(w_2) \tag{4}$$

where $S(w_1)$ and $S(w_2)$ are the set of synsets corresponding to w_1 and w_2, respectively.

A different method of computing the similarities between two words involves projecting them in a word space model [13]. This representation is composed of sparse semantic bits and it is also known as Distributional Memory [14]. Active bits can be interpreted as firing neurons in an analogy to neural networks. For each term we get a semantic fingerprint and in order to measure the similarity between them one can compute the cosine distance between the two semantic fingerprints. This has been implemented in the current work with the use of the Retina API [15]. However, this did not give results as good as in the case of the previous method, as it is described in the Section 3.

So far, both similarity computation methods target pairs of words only, but what we needed for this work was to compute similarity scores between a set of concepts corresponding to a certain image and a set of words corresponding to a certain topic description summarization. In order to take into account both the similarity between words and the confidence of every concept we summed up all similarity measures, weighted by the confidence of the respective concept for each concept-word pair. Therefore, for a certain image i and a certain topic j the similarity score is computed as follows:

$$score_{i,j} = \sum_{k} \sum_{l} d(w_k, w_l) \times c_k, \tag{5}$$

where $w_k \in C^{(i)}_{relevant}$, $w_l \in T^{(j)}$ and c_k is the confidence associated to w_k. Thus, we obtain high scores when there is both a high confidence for the given concept and a strong similarity between it and at least one word from the topic description summarization.

2.6 Similarity filtering

Finally, once we obtained the clusters and the similarity score between each image from the cluster and the topic description we select the best candidates for submitting a run. We remind that the task requested to submit a list of images for each topic representing images that are both relevant for the topic query and diverse. Therefore, we sorted the clusters in descending order based on the mean value of the similarity scores of the images that it contained. From the n-best ranked clusters we then selected the images with the highest similarity score and proposed them as the candidates for our run.

3 Experimental results

3.1 Results on development data

Our first approach was to compute the similarity score for each image-topic pair and just perform an ordering based on that score. Theoretically, images that contained objects in close connection, from a semantic point of view, with the topic description should be assigned a higher score. However, this was not exactly the case, as we saw from visual inspection of the highest ranked images, but for a few exceptions. We ran several tests to see how the data is distributed according to the similarity score. Knowing that the confidence scores are not evenly distributed among the images, as described in Section 2.1, we wanted to better understand the impact that low values of the confidence-similarity score products have on the final result. These values can also be considered noise added to the final score value. Therefore, we imposed several thresholds on this product. In other words, eq. 5 now becomes

$$score_{i,j} = \sum_{k}\sum_{l} d(w_k, w_l) \times c_k \times \mathbb{1}_{[d(w_k,w_l)\times c_k > th]}, \tag{6}$$

where th is an imposed threshold. Having obtained these results we plotted for each topic the confidence scores vs. the image's position number in the ordered shortlist only for the images from the ground truth that was provided by the organizers. In Fig. 3 and 4 we can see with different colors where the ground truth images are located in our system's output. Ideally, all the colored points should have been as close as possible to the left edge of the figure, namely they should have the highest confidence scores. We can see that the figures have a shape similar to a cotangent, but we would have desired for it to be steeper and concentrating more points on the left side.

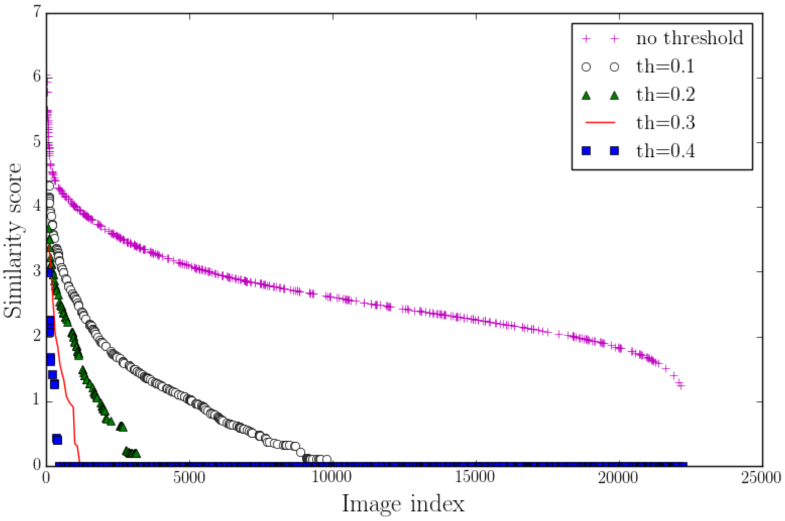

Fig. 3. Similarity scores obtained with the Wu-Palmer distance for topic T10.

We can also observe that by imposing thresholds we get a steeper shape as it was desired. Unfortunately, the concentration of images with a high confidence score (on the left side of the figure) drops. This is valid for both Wu-Palmer and cosine distance between semantic fingerprints.

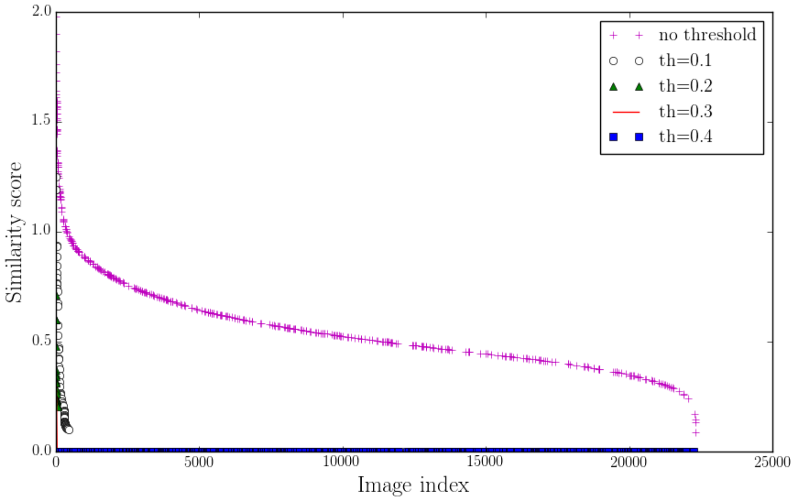

Fig. 4. Similarity scores obtained with the cosine distance for topic T10.

We conclude it is better to not impose any threshold for this type of confidence score computation. An explanation to this is that a significant part of the noisy concepts have already been eliminated when we imposed the threshold from eq. 1. This appears to be enough to reduce the noise in the final results.

3.2 Results on test data

However, the obtained results were not satisfactory so we needed to bring last minute modifications to our algorithm. We wanted to make use of the similarity score that we previously computed so the extension to the first approach came in the form of the filtering described in Section 2.6.

In more detail, we ran a hierarchical clustering algorithm on the shortlist of images corresponding to a certain topic and stopped it when it reached a total of 30 clusters. The next step was to sort these clusters according to the mean similarity score computed over the images belonging to each cluster. In order to submit 50 images for each topic we selected the 2 best ranked images from the first 25 clusters. The official results of the run can be seen in Fig. 5 and the last value to be considered in comparing the runs submitted by the participants is F1@10, as described in [5].

It is worth mentioning that the F1-measure at X was computed as the harmonic mean of the Precision at X and Cluster Recall at X. Our algorithm obtained various results depending on the topic, with the best values obtained for topics which contained an enumeration of items which may appear in a relevant picture or whose shortlist (Section 2.3) is very restrictive.

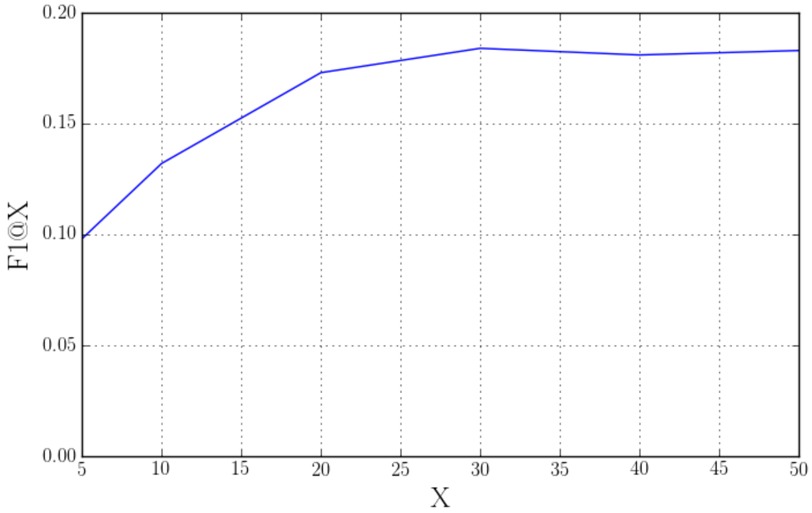

Fig. 5. F1@X measure official results.

4 Conclusions

We presented the results of our run to the ImageCLEF 2017 Lifelog Summarization subtask. The submitted run yielded results that were not satisfactory as they were not as good as those of the other participants. However, this work proposes a system which does not need any additional or external data and it heavily relies on the outputs of systems that were not tested under these conditions. We were surprised to see that for certain topics our system performed quite well while for others it was completely unreliable, which lead to an overall weak performance. Good performances have been obtained for topics which contained an enumeration of items that can be present in the image to make it relevant. Such an example is the topic related to shopping where there is an enumeration of the types of stores that make an image relevant for that certain query. Moreover, a good match between these items and the objects with high confidence identified by the concept detector, such as butcher shop, candy store or toyshop, also proved vital for the outcome of our system.

We identified a large series of tunable parameters which can improve the results such as the threshold from eq. 1, the similarity score computation mechanism, the number of clusters for the hierarchical clustering etc. Another weak point was the lack of correlation between the concepts output by the Caffe Concept Detector and the meaning of the textual descriptions of the topics as it is a daunting task to evaluate similarity between phrases which address completely different subjects. Having a concept detector trained for the specific task of recognizing items that we can encounter on a daily basis would offer more precise information.

Moreover, making use of the entire metadata linked to the database could have improved the results. On this point we remind that the temporal information has not been used. It is clear that once an image is certain to fit a certain query then it is very likely that at least one of the neighboring images before and after is also a viable candidate for that specific query. Also, special annotations or interpretation of the topic descriptions could have aided the similarity computation system.

To sum up, we think that this system offers an interesting perspective on how a simple clustering algorithm can benefit from a textual interpretation. Further work can address finding the right set of parameters and better understanding the mixture of visual and textual interpretation of multimedia content.

Acknowledgments. This work was funded under UEFISCDI research grant PN-III-P2-2.1-PED-2016-1065, agreement 30PED/2017, project Real-time IP Camera-based Intelligent Video Surveillance Security System with DROP Retrieval - SPOTTER.

References

1. Martin Dodge and Rob Kitchin. 'Outlines of a World Coming into Existence': Pervasive Computing and the Ethics of Forgetting. *Environment and Planning B: Planning and Design*, 34(3):431–445, 2007.
2. Cathal Gurrin, Alan F. Smeaton, and Aiden R. Doherty. Lifelogging: Personal big data. *Foundations and Trends in Information Retrieval*, 8(1):1–125, 2014.
3. Gareth J. F. Jones, Cathal Gurrin, Liadh Kelly, Daragh Byrne, and Yi Chen. Information access tasks and evaluation for personal lifelogs. In *EVIA@NTCIR*, 2008.
4. Cathal Gurrin, Hideo Joho, Frank Hopfgartner, Liting Zhou, and Rami Albatal. Overview of ntcir-12 lifelog task, 2016.
5. Duc-Tien Dang-Nguyen, Luca Piras, Michael Riegler, Giulia Boato, Liting Zhou, and Cathal Gurrin. Overview of ImageCLEFlifelog 2017: Lifelog Retrieval and Summarization. In *CLEF 2017 Labs Working Notes*, CEUR Workshop Proceedings, Dublin, Ireland, September 11-14 2017. CEUR-WS.org <http://ceur-ws.org>.
6. Bogdan Ionescu, Henning Müller, Mauricio Villegas, Helbert Arenas, Giulia Boato, Duc-Tien Dang-Nguyen, Yashin Dicente Cid, Carsten Eickhoff, Alba Garcia Seco de Herrera, Cathal Gurrin, Bayzidul Islam, Vassili Kovalev, Vitali Liauchuk, Josiane Mothe, Luca Piras, Michael Riegler, and Immanuel Schwall. Overview of ImageCLEF 2017: Information extraction from images. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction 8th International Conference of the CLEF Association, CLEF 2017*, volume 10456 of *Lecture Notes in Computer Science*, Dublin, Ireland, September 11-14 2017. Springer.
7. Zhibiao Wu and Martha Palmer. Verbs semantics and lexical selection. In *Proceedings of the 32Nd Annual Meeting on Association for Computational Linguistics*, ACL '94, pages 133–138, Stroudsburg, PA, USA, 1994. Association for Computational Linguistics.
8. Long Xia, Yufeng Ma, and Weiguo Fan. VTIR at the NTCIR-12 2016 Lifelog Semantic Access Task. In *Proceedings of NTCIR-12*, Tokyo, Japan, 2016.

9. H. L. Lin, T. C. Chiang, L.-P. Chen, and P.-C. Yang. Image Searching by Events with Deep Learning for NTCIR-12 Lifelog. In *Proceedings of NTCIR-12*, Tokyo, Japan, 2016.
10. George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November 1995.
11. ProtoGeo Oy. Moves, 2013. [Online, available at `https://moves-app.com/`; accessed 30-May-2017].
12. Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 886–893, Washington, DC, USA, 2005. IEEE Computer Society.
13. M. Sahlgren and Stockholms universitet. Institutionen för lingvistik. *The Word-Space Model: Using Distributional Analysis to Represent Syntagmatic and Paradigmatic Relations Between Words in High-dimensional Vector Spaces*. SICS dissertation series. Department of Linguistics, Stockholm University, 2006.
14. Marco Baroni and Alessandro Lenci. Distributional memory: A general framework for corpus-based semantics. *Comput. Linguist.*, 36(4):673–721, December 2010.
15. Cortical.io. Retina. [Online, available at `http://www.cortical.io/`; accessed 30-May-2017].