CERTH @ MediaEval 2012 Social Event Detection Task

Emmanouil Schinas, Georgios Petkos, Symeon Papadopoulos, Yiannis Kompatsiaris
CERTH-ITI
6^{th} Km. Charilaou-Thermis
Thessaloniki, Greece
{manosetro, gpetkos, papadop, ikom}@iti.gr

ABSTRACT

This paper describes the participation of CERTH in the Social Event Detection Task of MediaEval 2012. The task had three challenges that called for the detection of different types of events taking part in different locations. The approach proposed by CERTH is based on the use of a "same class" model, which is trained using data from the SED 2011 challenge and which predicts whether two images belong to the same event. The same class model is used to organize the images in a graph, on which a community detection algorithm is applied. At a final processing step, each cluster is classified as corresponding or not to a particular location and type of event in order to obtain the final detection result. In our best runs we achieve F-measure and NMI scores of 18.66 and 0.187 respectively for challenge 1, 74.64 and 0.674 for challenge 2 and 66.87 and 0.465 for challenge 3.

Categories and Subject Descriptors

H.3 [**Information Search and Retrieval**]: Miscellaneous

1. INTRODUCTION

In this paper we present the approach followed by CERTH at the MediaEval 2012 Social Event Detection task, which calls for the detection of social events of a specific type taking part in particular locations in a large collection of images. Details of the task are provided in [2].

2. PROPOSED APPROACH

The approach that we utilize is based on what may be termed the "same class" model. A same class model operates on a pair of items that are described by a set of features or modalities and predicts whether the two items belong to the same class. It takes as input the set of per modality dissimilarities between the two items and produces a binary label. In the case of the Social Event Detection task, each pair of items is a pair of Flickr images and the output is a prediction whether the two images belong to the same event. It should be noted that the same class model was previously used for event detection using a different approach [4, 3]. We train the same class model using data from the 2011 Social Event Detection challenge.

Once the same model has been learned, it can be used to group the images in a different collection. Our previous approach, presented in [3] would require computing the output of the same class model for each pair of the 167K images of the collection. Instead, we utilize an approach similar to that of [4]: for each item in the collection we find its nearest neighbors according to each modality and only compare it to them. We construct a graph where each image of the collection corresponds to a node and the existence of a link between a pair of nodes indicates that the same class model has predicted that the corresponding images belong to the same event. Eventually, the nodes of the graph are clustered using an efficient community detection algorithm, the Structural Clustering Algorithm for Networks (SCAN) [5].

The communities produced by SCAN correspond to candidate social events. This set is processed further by splitting the events that exceed a predefined time range into shorter events. Furthermore, each image that does not belong to any event forms a single-item event. After an attempt to merge these single-item events into larger clusters, if they are close enough in time and space, the results are added to the list of candidate social events.

At a final processing step, each candidate event is classified as relevant or not to each challenge. In order to achieve this, the items clustered together by SCAN are used to obtain an aggregate representation of each candidate social event: median geolocations and accumulated tf-idf vectors are computed for each cluster. At the same time, term models for the specific locations and event types are built. Our approach is based on [1]. In particular, we collect images from Flickr that either have geotags that associate the images to the locations of interest (for the location models) or are relevant to the type of event of interest. Additionally, we have collected a random collection of images, which is not focused to any particular location or type of event. For each term appearing in the title, tags or description of the images of each of the collections, we compute the probability of appearance as:

$$p(w|set) = \frac{N_w + \delta}{(\sum_w^n N_w) + \delta n}$$

where N_w is the number of occurrences of term w in the set, n is the number of different terms appearing in the set and δ is a small constant (typically set to 0.5) that is included to regularize the probability estimate (i.e. to ensure that a new term that does not appear in the set is not assigned a probability of 0). To determine the most important terms in the set for which we want to find the word model, we compute the ratio of the likelihood of appearance in the focus set over the likelihood of appearance in the reference set. That is, we compute $p(w|set_{target})/p(w|set_{ref})$. In order to

MediaEval 2012 Workshop, October 4-5, 2012, Pisa, Italy

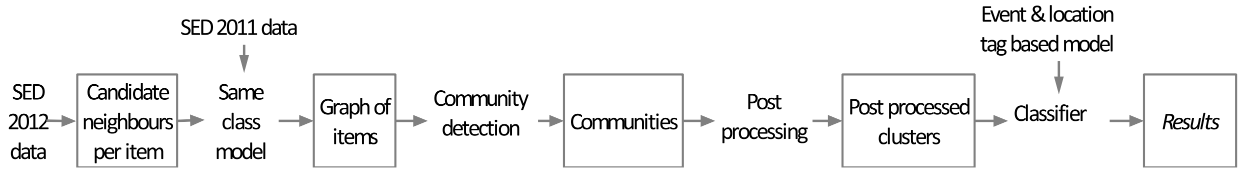

Figure 1: The approach utilized

Table 1: Results for the three challenges.

Challenge	Run	Precision	Recall	F-score	NMI
1	1	**59,12**	10,88	18,37	0,1599
	2	42,4	10,12	16,34	0,1545
	3	43,11	**11,91**	**18,66**	**0,1877**
2	1	85,57	66,19	**74,64**	**0,6745**
	2	**87,05**	65,07	74,48	0,6678
	3	81,41	**66,56**	73,24	0,6609
3	1	86,24	**54,61**	**66,87**	**0,4654**
	2	**88,43**	33,01	48,07	0,3984
	3	85,77	36,05	50,76	0,415

filter out the noisy terms in the target set, before the calculation of the ratio, we discard the terms whose number of occurrences in the target set is below a predefined threshold (10 occurrences). The terms after the filtering with the highest ratio should be the ones that are most related to the location or type of event that we want to describe. For each location or event type model we keep the 500 terms with the highest ratio. In order to classify each candidate event as being of a specific type or not and as being associated with a particular location or not, we compute the Jaccard similarity between the terms of its aggregated tf-idf vector and the set of selected terms for the corresponding event type or location model and if the similarity is above some threshold, the candidate event is included in the results. A schematic representation of our approach is illustrated in Figure 1.

3. EXPERIMENTS

We use a Support Vector Machine classifier in order to learn the same class model. The input features (and distance measures) used as input to the same class model are the following: user (1 if both images have been uploaded by the same user, 0 otherwise), title (BM25), tags (BM25), description (BM25), time taken (similarity is 1 if the time difference is below 12 hours, otherwise it is 0), GIST (euclidean distance) and SURF (aggregated using a VLAD scheme, similarity computed using euclidean distance). The same class model has been trained on data from the 2011 challenge.

Regarding the retrieval of the candidate neighbours of each item, the 50 nearest neighbours with respect to the textual features (title, description, tags) were considered, 150 with respect to time, 50 with respect to location (when it exists), 50 for GIST and 50 for SURF/VLAD. Efficient indexing schemes were utilized in order to rapidly determine the set of nearest neighbours for each modality.

The results can be found in Table 1. Each of the three runs of each challenge comprises a different graph of terms. In run 1, the graph is created by using a same class model trained with 10000 pairs of images. In run 2, a classifier trained with a larger dataset of 30000 pairs was used. In run 3, the same classifier as in run 2 is used, but a post processing step is applied to the community graph. More specifically, the nodes that do not belong to any of the detected communities are added to the community with which they have more edges, under the condition that this number is greater than two. Regarding the final processing step, for each challenge we used the same similarity threshold across the different runs. But for each challenge a different threshold was used for the Jaccard similarity, because of the different distribution of dominant terms in these challenges. Regarding the post processing step, for each challenge different parameters of time and space are used for the splitting and merging of the candidate events.

4. DISCUSSION

As it is evident from Table 1, better results were obtained for the second and third challenges. Moreover, moving from a smaller (run 1) to a larger (runs 2 and 3) training dataset for the same class model does not seem to improve most of the performance measures. This could be a result of overfitting and will have to be further investigated. Additionally, there seems to be a small improvement from run 2 to run 3 for recall.

In the future we will attempt to further refine the presented approach by training the same class model with a richer set of data that is representative of a wider class of events. Moreover, we plan to explore different graph construction strategies and community detection algorithms.

5. ACKNOWLEDGMENTS

This work is supported by the SocialSensor FP7 project, partially funded by the EC under contract number 287975.

6. REFERENCES

[1] B. O'Connor, M. Krieger, and D. Ahn. Tweetmotif: Exploratory search and topic summarization for twitter. In W. W. Cohen and S. Gosling, editors, *ICWSM*. The AAAI Press, 2010.

[2] S. Papadopoulos, E. Schinas, V. Mezaris, R. Troncy, and I. Kompatsiaris. Social Event Detection at MediaEval 2012: Challenges, Dataset and Evaluation. In *MediaEval 2012 Workshop*, Pisa, Italy, Oct 4-5 2012.

[3] G. Petkos, S. Papadopoulos, and Y. Kompatsiaris. Social event detection using multimodal clustering and integrating supervisory signals. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, ICMR '12, pages 23:1–23:8, NY, USA, 2012.

[4] T. Reuter and P. Cimiano. Event-based classification of social media streams. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*, ICMR '12, pages 22:1–22:8, New York, NY, USA, 2012.

[5] X. Xu, N. Yuruk, Z. Feng, and T. A. J. Schweiger. Scan: a structural clustering algorithm for networks. In *Proceedings of the 13th ACM SIGKDD*, KDD '07, pages 824–833, NY, USA, 2007. ACM.