Social Event Detection with Clustering and Filtering

Yanxiang Wang
Australian National University
u4950984@anu.edu.au

Lexing Xie
Australian National University
lexing.xie@anu.edu.au

Hari Sundaram
Arizona State University
hari.sundaram@asu.edu

ABSTRACT

We present a clustering and filtering approach for the Social Event Detection task in MediaEval 2011. Our algorithm makes use of time, location, as well as textual and visual features. We cluster the multimedia documents followed by retrieval-based filtering with partial event properties.

1. INTRODUCTION

The Social Event Detection (SED) [5] task at MediaEval 2011 present a challenging problem for retrieving and organizing social media around real-world events, such as sports games or events at a given concert venue. A key difference between the SED problem and earlier work on media event detection is that information about the target events are partially specified (via venue or type of sport), rather than completely unspecified [1, 2, 7] or specified for *each event* with examples [3].

Such problem specification motivate us to adopt a hybrid clustering and filtering approach. We first cluster the dataset with approaches similar to Becker [1] and Papadopoulos [6], tuned using a separate training set. We then filter the resulting clusters, using retrieval approaches on time, location, text and visual information.

2. APPROACHES

Since the SED task only provided an evaluation dataset [5], we compile a separate training collection using a subset of the upcoming dataset [1] with additional random photos from Flickr. To mimic the challenge proposed by SED2011, the training subset only contains upcoming events that are sports and music. The random photos added are within the same timeframe of the existing events. The performance of the algorithm is evaluated against ground-truth events in upcoming using F1.

The overall flow of our algorithm is shown in Figure 1. We perform two clustering phases before the filtering step.

2.1 Clustering on data set

We use a single-passed incremental clustering algorithm [1] to cluster the data. The similarity metrics used for each of the time-stamp, location, tags, textual features are as follow:

- **Time-stamp**: We represent time value as the minutes elapsed since the beginning of Unix epoch. If two times are more then a week apart, their similarity is 0. Otherwise, the similarity between two time-stamps t_1 and t_2 is computed as $s_t = 1 - \frac{t_1 - t_2}{t_w}$, where t_w as number of minutes in a week.

- **Location**: We compute the great circle distance (GCD)[1] between a pair of locations using the GeoPy library[2]. We set the location similarity s_l to 0 if the GCD value is greater than 50 miles, otherwise $s_l = 1 - \frac{GCD}{50}$.

- **Tags**: We use the Jaccard index[3] as the similarity s_g between two tag set.

- **Text**: We obtain a term-frequency vector from the photo title and description after stemming and eliminating the stop words. The cosine similarity is used as the text similarity s_w.

In clustering phase C2, we use a weighted combination of similarity functions $s' = w_g s_g + w_w s_w + w_l s_l$. We use $w_g = 0.65$, and $w_l = 0.15$, $w_w = 0.2$ if location data is available for both photos, otherwise $w_w = 0.35$. The centroid of each cluster is maintained in the end of the clustering step for filtering.

2.2 Retrieve relevant events cluster

In the first phase of filtering step, we remove the clusters outside the specified time and location constraints.

We subsequently filter the clusters with text and tags associated with the query term. We generate a text vector and a tag vector for each query term. We construct the two vectors via two Flickr API[4] methods. To construct the text vector, we call method *flickr.photos.search* with the query term. We build the text vector by normalizing text content from 100 most relevant results. Similarly, we call method *flickr.tags.getClusters* with the query term, to retrieve a set of tags statistically associated with the query term.

We use weighted combination similarity function described in 2.1 to compute the similarity between each centroid and

MediaEval 2011 Workshop September 1-2, 2011, Pisa, Italy

[1] http://en.wikipedia.org/wiki/Great-circle_distance
[2] http://code.google.com/p/geopy/
[3] http://en.wikipedia.org/wiki/Jaccard_index
[4] http://www.flickr.com/services/api/

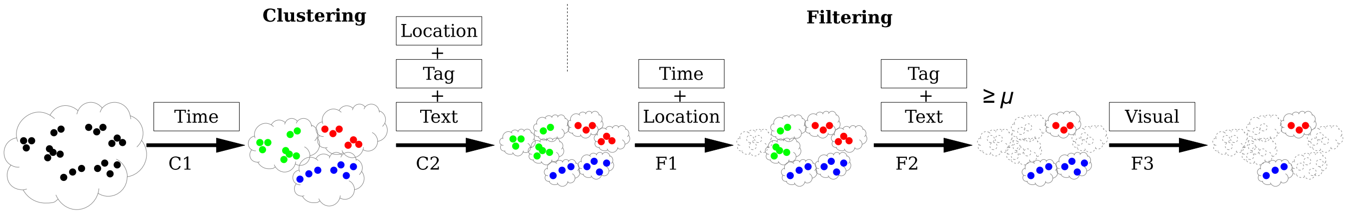

Figure 1: Overview of the clustering and filtering steps.

Run No.	**1**	**2**	**3**
Metrics	μ**:0.2**	μ**:0.1**	μ**:0.05**
Precision	12.53	62.88	84.86
Recall	58.79	52.93	52.54
F1	20.65	57.48	64.9
NMI	0.1166	0.2207	0.2367

Table 1: challenge 1 result

Run No.	**1**	**2**	**3**	**4**
Metrics	μ**:0.2**	μ**:0.1**	μ**:0.05**	μ**:0.1 last.fm**
Precision	38.5	59.26	66.89	56.16
Recall	66.34	43.9	6.04	18.9
F1	48.72	50.44	11.07	28.28
NMI	0.2941	0.448	0.2705	0.4491

Table 2: challenge 2 result

the query document. We specify a threshold μ to filter the clusters below the minimum similarity.

In F3, clusters are filter based on their visual information. we use a visual classifier [8] to label all photos in each cluster. We manually construct key, value pairs to represent the invalid class labels and corresponding threshold. A cluster is discarded if the fraction of photos with invalid label in cluster is greater than the threshold value.

3. RESULTS

For challenge 1, we feed search term 'Barcelona', 'soccer' and 'Rome', 'soccer' to the Flickr API method and perform three runs with different setting of μ shows in Table 1.

For challenge 2, in addition to the runs from search term 'Paradiso' and 'Parc del Forum', we take the idea of Liu [4]. We construct the tag set and text vector from artists' names, title and descriptions for each event found on last.fm[5] event directory to anchor a supplementary run 2.

While our results show promise, they can be substantially improved. However, the best performing result with $\mu = 0.1$ for F1 evaluation is still in the acceptable level. The results show that our recall value on average is lower than precision. Thus, in future work, we will further investigate to refine the filtering method to improve the recall value. Possible directions include: other tag and text construction strategy, augment visual filtering etc. To tackle the low performance on NMI value, we will study the clustering results to gain more insight.

[5] http://www.last.fm/api

4. REFERENCES

[1] H. Becker, M. Naaman, and L. Gravano. Learning similarity metrics for event identification in social media. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 291–300, 2010.

[2] L. Chen and A. Roy. Event detection from flickr data through wavelet-based spatial analysis. In *Proceeding of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 523–532, 2009.

[3] C. S. Firan, M. Georgescu, W. Nejdl, and R. Paiu. Bringing order to your photos: event-driven classification of flickr images based on social knowledge. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 189–198, 2010.

[4] X. Liu, R. Troncy, and B. Huet. Finding media illustrating events. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ICMR '11, pages 58:1–58:8, 2011.

[5] S. Papadopoulos, R. Troncy, V. Mezaris, B. Huet, and I. Kompatsiaris. Social Event Detection at MediaEval 2011: Challenges, Dataset and Evaluation. In *MediaEval 2011 Workshop*, Pisa, Italy, September 1-2 2011.

[6] S. Papadopoulos, C. Zigkolis, Y. Kompatsiaris, and A. Vakali. Cluster-based landmark and event detection for tagged photo collections. *IEEE MultiMedia*, 18:52–63, 1 2011.

[7] T. Rattenbury, N. Good, and M. Naaman. Towards automatic extraction of event and place semantics from flickr tags. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 103–110, 2007.

[8] J. R. Smith and et al. IBM multimedia analysis and retrieval system. http://www.alphaworks.ibm.com/tech/imars.