TUW @ Retrieving Diverse Social Images Task 2014

João R. M. Palotti
Vienna University of Technology
palotti@ifs.tuwien.ac.at

Navid Rekabsaz
Vienna University of Technology
rekabsaz@ifs.tuwien.ac.at

Mihai Lupu
Vienna University of Technology
lupu@ifs.tuwien.ac.at

Allan Hanbury
Vienna University of Technology
hanbury@ifs.tuwien.ac.at

ABSTRACT

This paper describes the efforts of Vienna University of Technology (TUW) in the MediaEval 2014 Retrieving Diverse Social Images challenge. Our approach consisted of 3 steps: (1) a pre-filtering based on Machine Learning, (2) a re-ranking based on Word2Vec, and (3) a clustering part based on an ensemble of clusters. Our best run reached a F@20 of 0.564.

1. INTRODUCTION

Diversification is an interesting problem for the information retrieval community, being a challenge for both text and multimedia data. Focused on image retrieval, the MediaEval 2014 Retrieving Diverse Social Images Task [1] was proposed to foster the development and evaluation of methods for retrieving diverse images of different point of interest.

2. METHODS

We employed a distinct set of methods for each run. Here we explain all the approaches and in Table 1 we show the combinations used for each run.

2.1 Pre-Filtering

We employed a pre-filtering step to exclude likely irrelevant pictures. The goal of this step is to increase the percentage of relevant images. We studied two approaches: (1) a filtering step based on a simplified version of Jain et al. [2] experiments, removing images without any view, geotagged more than 8 kilometers away from the point of interest (POI) and with a description length longer than 2000 characters; (2) we trained a Logistic Regression classifier on the whole 2013 and 2014 data, using as features the ones described above and also the images' license, the time of the day (morning, afternoon, night) and the number of times the POI appeared in the title and descriptions of an image.

2.2 Re-ranking

For re-ordering the results, we used the title, tags and description of the photos. For text pre-processing, we decomposed the terms using a greedy dictionary based approach. In the next step, we expand the query using the first sentence of Wikipedia which helped for place disambiguation. We tested four document similarity methods based on Solr[1], Random Indexing[2], Galago[3] and Word2Vec[4][4]. Among all, we found the best result using a semantic similarity approach based on Word2Vec.

Word2Vec provides vector representation of words by using deep learning. We trained a model on Wikipedia and then used the vector representation of words to calculate the text similarity of the query to each photo.

Apart from the Word2Vec scores, we extracted binary attributes based on Jain et al. [2], as we did in the pre-filtering step, and we used a Linear Regression to re-rank the results based on the development data.

2.3 Clustering

We worked on three methods for clustering, all based on similarity measures. They share the idea of creating a similarity graph (potentially complete) in which each vertex represents an image for one point of interest, and each edge represents the similarity between two images. Different similarity metrics and different set of features can be used. Next, we explain each algorithm and how we combined them.

2.3.1 Metis

The first approach, called Metis [3], tries to collapse similar and neighbor vertices, reducing the initial graph to a smaller one (known as coarsening step). Then, it divides the coarsest graph into a pre-defined number of graphs, generating the clusters.

2.3.2 Spectral

Spectral clustering [5] can also be seen as a graph partitioning method, which measures both the total dissimilarity between groups as well as the total similarity within a group. We used the Scikit-learn[5] implementation of this method.

2.3.3 Hierarchical

Hierarchical clustering [6] is based on the idea of a hierarchy of clusters. A tree is built in a way that the root gathers all the samples and the leaves are clusters with only one sample. This tree can be built bottom-up or top-down. We used the bottom-up implementation from Scikit-learn.

MediaEval 2014 Workshop, October 16-17, 2014, Barcelona, Spain

[1] http://lucene.apache.org/solr/
[2] https://code.google.com/p/semanticvectors/
[3] http://sourceforge.net/p/lemur/galago/
[4] https://code.google.com/p/word2vec/
[5] http://scikit-learn.org/

Table 1: Each run and its settings.

Run	Pre-Filtering	Re-Ranking	Clustering	Credibility
1	Based on [2]	-	Combined on HOG,CN3x3,CN	-
2	-	Word2Vec	Metis on Text Similarity	-
3	-	Word2Vec	Combined on HOG,CN3x3,CN	-
4	-	Word2Vec	Combined on HOG,CN3x3,CN	ML to remove elements
5	Based on ML	Word2Vec	Combined on HOG,CN3x3,CN	ML to re-rank elements

Table 2: All results - the best run according to the official metric was Run1 reaching a F@20 of 0.564.

Run	2014 Development Set						2014 Test Set					
	P@10	CR@10	F1@10	P@20	CR@20	F@20	P@10	CR@10	F1@10	P@20	CR@20	F@20
1	0.827	0.282	0.416	0.805	0.465	0.585	0.798	0.283	0.412	0.769	0.450	0.560
2	0.903	0.262	0.400	0.870	0.425	0.564	0.806	0.251	0.377	0.773	0.381	0.501
3	0.870	0.301	0.444	0.813	0.483	0.601	0.794	0.281	0.410	0.744	0.449	0.553
4	0.890	0.297	0.441	0.827	0.503	0.619	0.806	0.280	0.412	0.754	0.443	0.552
5	0.837	0.299	0.435	0.792	0.478	0.588	0.780	0.276	0.403	0.729	0.444	0.546

2.3.4 *Merging*

We found that the clustering methods were unstable as modifications in the filtering step caused a great variation in the clustering step. Therefore, we decided to implement a merging heuristic, which takes into account different points of view from each clustering method and/or feature set, being potentially more robust than using one single algorithm.

Given c different clustering algorithms, f different feature sets, and m distance measures, there are $c \times f \times m$ possible cluster sets. In our work, we used the 3 algorithms described, 3 features sets (HOG, CN, CN3x3, see [1] for details about these features) and 2 distance measures (cosine and Chebyshev), giving us 18 different cluster sets for each POI. We can then compute a *frequency matrix* that will hold for every two documents the number of cluster sets in which they occur in the same cluster. Next we create a re-ranked list of the images from the original list (Flickr ranking) based only on this frequency matrix. In order to do that, we define a threshold t and a function F on a set of frequencies that determine when a document should be moved to the re-ranked list. Suppose the re-ranked list contains the documents $D_1, ..., D_i$ and we want to know if a document D_k in the original list can be moved to the re-ranked list at position $i+1$. We compute the frequencies $f_1, ..., f_i$ between D_k and each $D_1, ..., D_i$ and if $F(f_1, ..., f_i) < t$, D_k is moved to the re-ranked list, otherwise it is not. After all the elements in the original list are processed, if there are still remaining documents not moved to the re-ranked list, the value of t is increased and these documents are reprocessed. The algorithm continues until there are no documents left in the original list. The functions for F used in this work were maximum, minimum and mean, but other measures, such as mode, median or any percentile could be easily employed as well.

2.4 Credibility

Our approaches were based on Machine Learning (ML): we trained a Logistic Regression classifier to learn if a document was relevant or not based on the credibility data (used only face proportion, location similarity, upload frequency and bulk proportion). We tested two methods: (1) excluding documents set as irrelevant for Run4 and (2) moving to the bottom of the list irrelevant documents for Run5.

3. EXPERIMENTS

We submitted all 5 runs, varying on the use of pre-filtering, the re-ranking method, the clustering approach and the use of credibility. Details are shown in Table 1 and the results are shown in Table 2. Based on the development data, we were expecting Run3 and Run4 to be our best runs, but the results on the test data shows that we probably overfitted the development set for the re-ranking and credibility part. The best result was that the cluster ensemble proved to be robust for this task.

4. CONCLUSION

Our experiments show that an ensemble of clusters can be a robust way to diversify results. Unfortunately our re-rank method did not work in the test set as well as it did in the development set. Last, the use of credibility also seems to have overfitted the development data, not being effective for the test set.

ACKNOWLEDGMENTS.

This research was funded by the Austrian Science Fund (FWF) project number I1094-N23 (MUCKE) and project number P25905-N23 (ADmIRE).

5. REFERENCES

[1] B. Ionescu, A. Popescu, M. Lupu, A. L. Gînscă, and H. Müller. Retrieving Diverse Social Images at MediaEval 2014: Challenge, Dataset and Evaluation. In *MediaEval 2014*, 2014.

[2] N. Jain, J. Hare, S. Samangooei, J. Preston, J. Davies, D. Dupplaw, and P. H. Lewis. Experiments in diversifying flickr result sets. In *MediaEval 2013*, 2013.

[3] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.*, 1998.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *CoRR*, 2013.

[5] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2000.

[6] J. H. Ward. Hierarchical grouping to optimize an objective function. *JASA*, 1963.