TUW@Retrieving Diverse Social Images Task

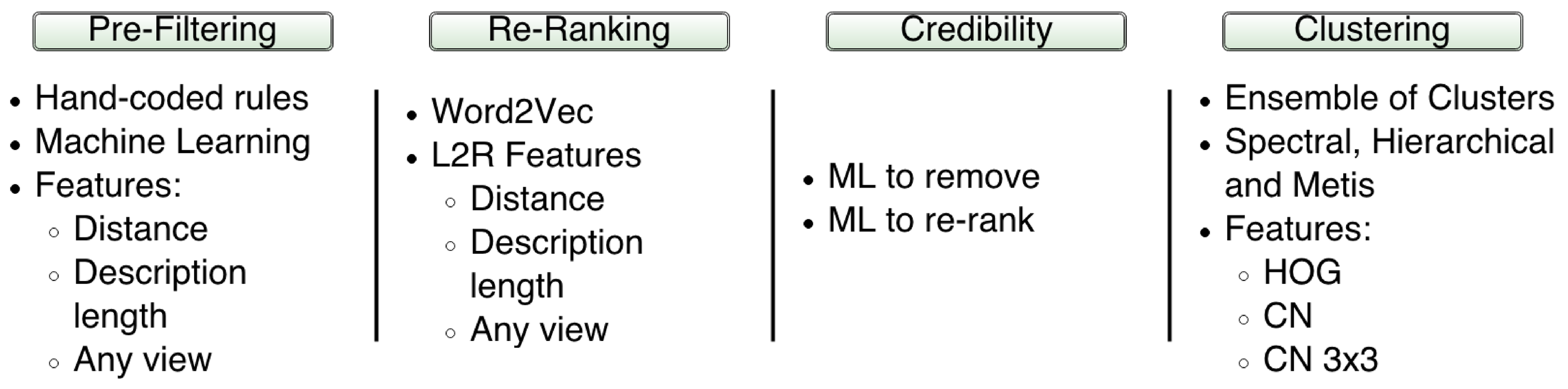
Working Notes

Joao Palotti* & Navid Rekabsaz & Mihai Lupu & Allan Hanbury

Vienna University of Technology

palotti@ifs.tuwien.ac.at — ifs.tuwien.ac.at/∼palotti

Overview



Pre-filtering

We employed a pre-filtering step to exclude likely irrelevant pictures, increasing the percentage of relevant images. We studied two approaches:

1. Static rules to remove 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. Machine Learning (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

Re-ranking

We tested four document similarity methods based on Solr, Random Indexing, Galago and Word2Vec. The best performance was found using Word2Vec, which provides a 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 as we did in the pre-filtering step, and we used a Linear Regression to re-rank the results based on the development data.

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.

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.

Metis

The method called Metis tries to collapse similar and neighbor vertices, reducing the initial graph to a smaller one (coarsening step). Then, it divides the coarsest graph into a pre-defined number of graphs, creating the clusters.

Spectral

Spectral clustering 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.

Hierarchical

Hierarchical clustering 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.

Ensembling

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.

We consider that each clustering algorithm can be used with different sets of features and similarity measures: **Cluster**(*Algorithm, Distance Metric, Feature Set*). The main measure used for re-ranking the original list is the frequency that two images are set to the same cluster in the ensemble.

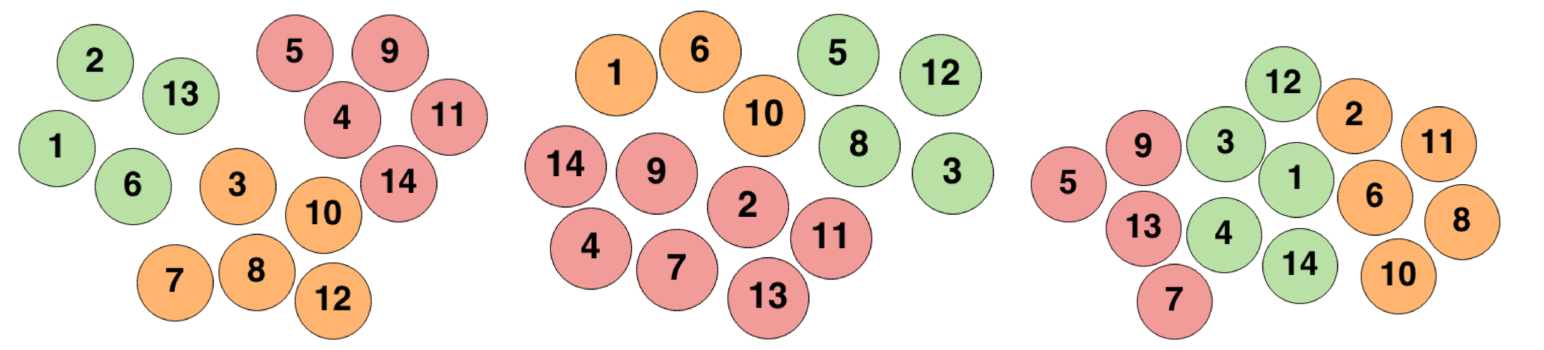


Figure 1: Three different ways to cluster 14 images. Different clustering algorithms, similarity measures and feature sets can strongly change the clustering assignments

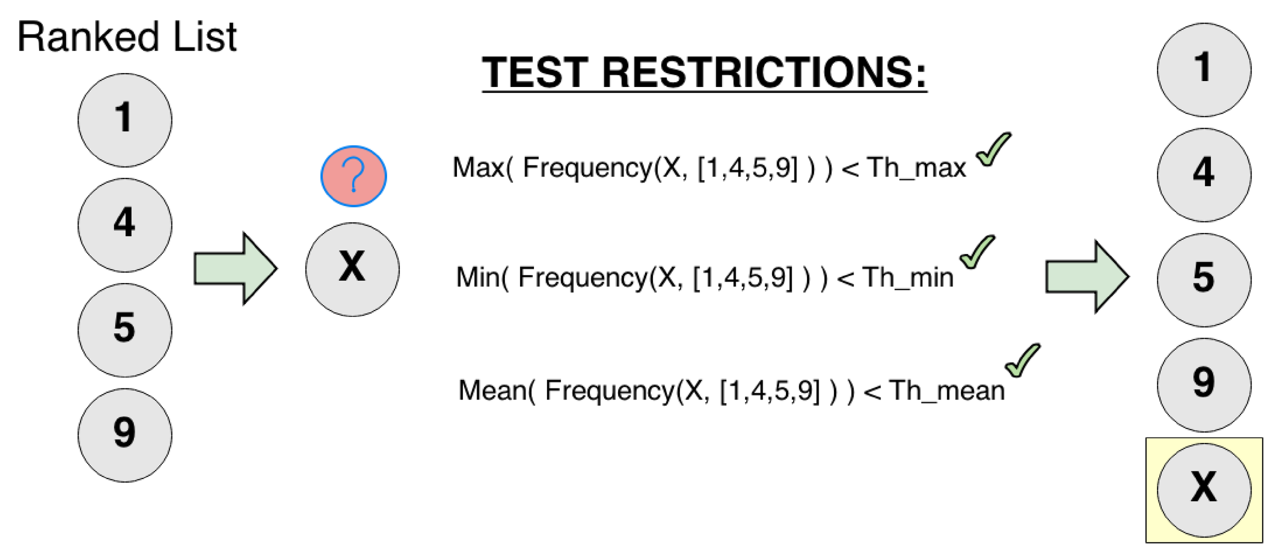


Figure 2: Illustrates how simple frequency-based rules can be effectively used to decide if an element should be added to the final re-ranked list.

Results

The configuration of our 5 runs and their results were:

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

Run 1 was the best one, reaching a F@20 of 0.56.

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

Conclusions

- Ensemble of clusters is a simple and very robust way to diversify results.
- Few data for Machine Learning methods may be the main reason for overfitting the development set (and not improving for Runs 2,3,4 and 5)
- Future: evaluate the clustering procedure in other scenarios (e.g., text).

Acknowledgements

This research was funded by the Austrian Science Fund (FWF) project number I1094-N23 (MUCKE) and project number P25905-N23 (ADmIRE). The authors would like also to thank the organizers for their excellent work.