Visual Reranking for Image Retrieval over the Wikipedia Corpus

Débora Myoupo* Adrian Popescu† Hervé Le Borgne*
Pierre-Alain Moëllic*
*CEA LIST, Multilingual Multimedia Knowledge Engineering Laboratory,
18, route du panorama, BP6 FONTENAY AUX ROSES, F-92265 France.
debora.myoupo@gmail.com herve.le-borgne,pierre-alain.moellic@cea.fr
†Telecom Bretagne, Technople Brest-Iroise, 29238 Brest
adrian.popescu@telecom.eu

Abstract

This paper describes the approach we developed for the WikipediaMM task on 2009 [4], which builds on our last year contribution. The main novelties are the refinement of textual query expansion procedure and the introduction of a k-NN based visual reranking procedure. Our main purpose was to test whether combining textual and content based retrieval improves over purely textual search and the results we report here confirm that combining modalities results in a significant improvement of results.

Keywords

Image retrieval, query expansion, Wikipedia, CBIR, image reranking, k-NN.

1 Introduction

The wikipediaMM task consists of retrieving images from a large-scale collection of heterogenous images [5, 4]. We propose a fully automatic approach which first retrieves images for a given query using textual query expansion and then reranks the results of the textual run using visual representations of the queries. Query expansion exploits the categorical structure of Wikipedia in order to find and rank relevant concepts from the collaborative encyclopedia. Terms in the query are compared to Wikipedia categories and articles which matched a maximum number of such terms are ranked best. We build visual models for each topic by retrieving Web images which match that topic from Google Image and Yahoo! Image and reranking them using a k-NN approach and an negative set of images which contains diversified images. Each Web image is compared to all other images retrieved with the same query and to the negative set and is well ranked if it is visually distant from the external class. Since Web images are usually noisy, we retain only the best ranked images for creating the visual model. Then, each image retrieved from the Wikipedia corpus is compared to the visual model and to the negative set in order to favor those which are close to the visual model. We index images using both local (bags of visual words) and global (texture-color) descriptors and report results for both types of indexers as well as for their fusion. There are two main common points to these steps. The first is that we use an intermediate *conceptual level* as a reference to describe the images. The second is that in both cases, we use external sources (from the web) to build these conceptual level.

2 Automatic building of conceptual structure

Last year, we introduced a Wikipedia based query expansion procedure which takes advantage of the categorical structure of the encyclopedia[2] and we obtained encouraging results. This year we refine our approach by modifying the way query related concepts are ranked.

2.1 Data sources

The main resource we exploited was Wikipedia. Dumps of the encyclopaedia are regularly provided for a free use. We downloaded the April 2009 English dump, which contains over 2.6 million articles and is provided as a single file, in XML format. Next, we split the dump into individual articles in order to process the information faster. The information in Wikipedia spans over a large number of conceptual domains, with a high number of articles describing known people, places, entertainment, organisations animals and plants. Each article is placed in at least one category, a property that facilitates the extraction of *IsA* relations from the encyclopedia.

2.2 Conceptual neighbourhood building

Wikipedia images are accompanied by brief textual descriptions and query expansion is an appealing way to improve recall and, if performed in a judicious way, to also improve results precision. Topics are preprocessed in order to eliminate stop words and eliminate visual terms from a closed list (including image(s), photograph(s) etc.). Then, we lemmatize remaining terms and arrange them using their term frequency in Wikipedia in order to favor rare terms (which are more likely to be discriminant than frequent words). Then, we compare terms in the query to Wikipedia categories and retain all articles that match at least on term in the query. A limit of 5000 articles is imposed to speed up the processing. For instance, a query with *orthodox icons with Jesus* will have related concepts such as *Christ the Redeemer* and *Theotokos* but also *Our Lady of Kazan* or *Manhattan*.

It it obvious that these concepts are not equally relevant to the query and it is necessary to rank them so as to put the most pertinent first. Whereas *Christ the Redeemer* and *Theotokos* match all terms in *orthodox icons with Jesus*, *Our Lady of Kazan* matches only *orthodox icons* and *Manhattan* is found because *orthodox* appears in one of its categories (*United States Places with Orthodox Jewish communities*). We rank concepts by counting the number of terms from the initial query which appear in each article's categories and by favoring related concepts which match rare terms in the query. At this point, there are usually several concepts with the same score and in order to differentiate them we refine the ranking by answering the following questions:

- is the concept ambiguous?
- do all terms in the initial query appear in the first paragraph of the Wikipedia article?
- do all terms in the initial query appear in the article's text?

The refined list of related concepts will favor unambiguous elements and concepts which contain all terms in the query either in the first paragraph of the associated article (which is often a definition) or in the remaining text.

3 Textual retrieval

Relevant images are found by launching queries with the initial terms and the expanded queries in the following order:

- all terms in the initial queries and related concepts
- the initial query

- a part of the initial queries and related concepts
- related concepts or a part of the initial query

The intuition behind this type of querying is that an image described by many terms (from the initial query and the related concepts) is more likely to be relevant than another image described by fewer terms. Weights are applied to the terms in the initial query and to related concepts and individual image scores are calculated by adding these scores.

Let $\mathcal{R}_t$ be the list of images returned by the textual matching procedure.

4 Visual reranking

The basic idea of our content based reranking procedure is that an image which is visual close to the visual model of a query is more likely to be a good answer than another image which is less similar to the visual model. To evaluate the similarities between the $\mathcal{R}_t$ images and a topic, we need to obtain a low-level description of that particular topic. We create a *visual model* (a *positive set* of images $\mathcal{R}_{pos}$ which depicts the concept) using Web images downloaded from Google Image and Yahoo! Image. A *negative set* $\mathcal{R}_{neg}$ containing diversified images is manually constructed and used as an outlier for all topics in order to discard images which are not visually close to the topic's visual model. Then we use the *visual coherence* to evaluate the relevance of both sets ($\mathcal{R}_{all} = \mathcal{R}_{pos} \cup \mathcal{R}_{neg}$) and rerank them. Eventually, we merge this new ordered list $\mathcal{R}_t$ using *window merging* or *blocks merging*.

4.1 Visual coherence

The *visual coherence* of an image is a metric mesuring the similarities beetween an image and a concept. This metric is computed using two sets of images : a *positive set* $\mathcal{R}_{pos}$ containing N_{pos} various relevant images of the concept and a *negative set* $\mathcal{R}_{neg}$ of N_{neg} non relevant images. These two sets compose the *visual prototype* of the concept. We compute the *visual coherence score* as a couple of scores :

False neighbours: For an image, we search for its N_{neigh} closest neighbours in $\mathcal{R}_{pos}$ as well as its N_{neigh} closest neighbors in $\mathcal{R}_{neg}$. The visual similarity is calculated using a low-level descriptor (global or local) and the euclidean distance. The two lists of N_{neigh} neighbours are then merged and the first part of the VC score is defined as the number of neighbours which belong to the negative set among the first N_{neigh} images of this $2 \times N_{neigh}$ size list. In an ideal case, an image which perfectly represents the concepts will not have any pictures of $\mathcal{R}_{neg}$ among its N_{neigh} closest neighbours. Inversely, the more negative elements among the first N_{neigh} neighbours are found, the less the image is a good representative of the concept.

Distances: The second score is the sum of the distances of their N_{sum} closest neighbours in $\mathcal{R}_{pos}$. A small value of this sum implies that the image is visually similar to the concept in the descriptor's space. The distances computation is based on the same descriptor as in the previous step.

This algorithm is strongly dependant to the descriptor and the metric.

To rank a set of images according to their likeness to a concept, we sort them using only the first score (which is a number between 0 and N_{neigh}). For two images having the same number of neighbors, we use the second score of the visual coherence to refine the ranking.

4.2 Visual prototype conception

We hypothesize that a visual prototype of a topic can be extracted by querying a web-scale image search engine with that particular topic and by reranking the answers in order to reduce noise. N_q images. We also download a set of N_{neg} various images that we index to build the *negative set*

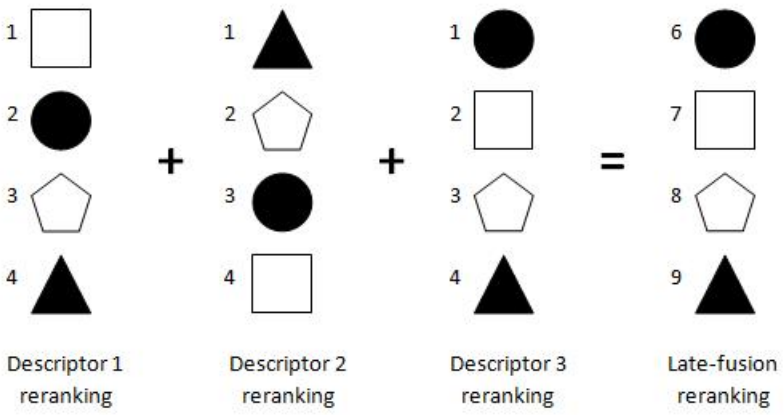

Figure 1: Scheme of the Late-fusion Reranking process

$\mathcal{R}_{neg}$. The *negative set* contains diversified images which depict a large number of concepts such as *mountain*, *dog*, *car*, *football* or *protest*. Several content-based descriptors are then computed from the *raw set* of images. For the visual prototype to be effective, the retained images need to be as accurate as possible and it is necessary to filter out noisy results returned by the Web search engine. For this, we compute the *VC* on the raw set: for each image, the $N_q - 1$ other images of the set are temporarily considered as $\mathcal{R}_{pos}$ and used with $\mathcal{R}_{neg}$ to compute its VC score.
We keep the top N_{pos} results only, that are considered good enough to represent the concept and be part of its visual prototype.
Since the visual coherence computation depends on the features, a visual prototype is thus also defined for each descriptor. It is finally composed of an ordered list of N_{pos} *positive images* and a set of N_{neg} *negative images* relatively to the concept.

4.3 Textual results reranking

Let $\mathcal{R}_t$ be the list of images returned by the textual matching procedure. It can now be reordered using the visual prototypes using the same k-NN method used to build the prototype itself. In practice, we compute the signature of each image of $\mathcal{R}_t$ and its distance to all the images of the visual prototype (both to the $\mathcal{R}_{pos}$ and the $\mathcal{R}_{neg}$ set) and compute their visual coherence. Since the visual prototype covers different aspects of the topic, we expect that a relevant result from the textual run to be related to some of the images composing the visual prototype. Since the visual prototype is dependant on the descriptor, we run experiments using:

Descriptor Reranking One descriptor is chosen. We use the visual prototype created with this descriptor to compute the visual coherence scores of the $\mathcal{R}_t$ list. Then, the list is simply ranked according to the VC score.

Late-fusion Reranking The picture is reordered according to the sum of its ranks into the lists coming from several Descriptor Rerankings (i.e. using several descriptors) (cf. figure 1).

Early-fusion Reranking We fuse the visual prototypes which depict the same concept but use two different descriptors by crediting each image with the sum of its rank. Then, the new visual prototype is used to build each Descriptor Reranking. The Descriptors Rerankings are merged using the Late-fusion Reranking approach.

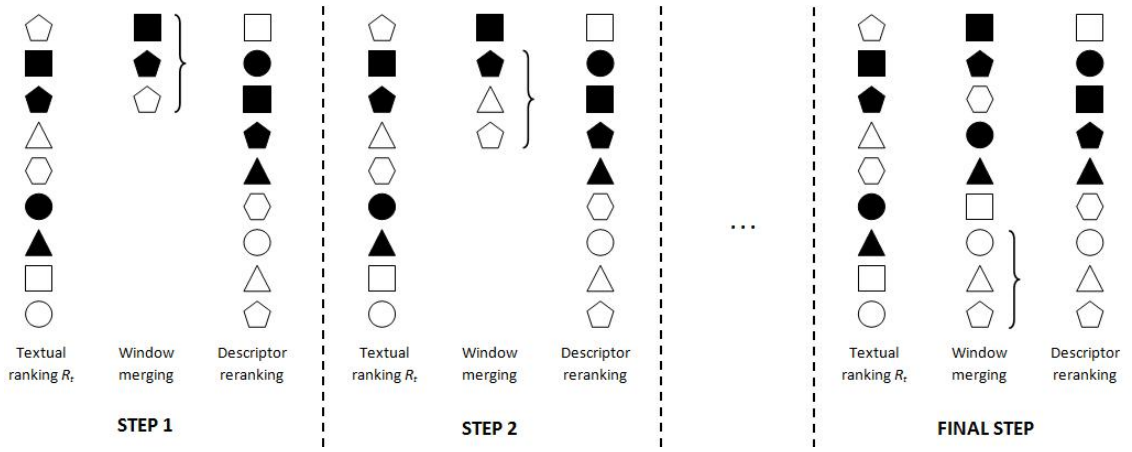

Figure 2: Scheme of the Window merging process

4.4 Textual and visual information merging

Textual results have weights which express their closeness to the topic. We describe two methods for merging textual and visual information.

4.4.1 Window merging

In this process, we consider two rankings : $\mathcal{R}_t$ and a content-based ranking (Descriptor Reranking, Late-fusion Reranking, Early-fusion Reranking). A window of S_{window} images slides on the textual order:

1. All the elements in the window are reranked according to the content based order.
2. The first one is selected as a new member of the window merging order.
3. Next, it is removed from the window.
4. Then, we add the next element of $\mathcal{R}_t$ in the window.

We repeat the process until all the elements of $\mathcal{R}_t$ are ranked (cf. figure2).

4.4.2 Blocks merging

The $\mathcal{R}_t$ ranking is divided into blocks and each block is reranked accordind to the same content-based order. A block is composed of images which are similarly related to the topic. For instance described by all terms in the initial query and a related concept or described only by all the terms is the initial query. Given a query with *orthodox icons with Jesus*, two images which are annotated with all the terms in the query and with *Christ the Redeemer*, respectively *Theotokos* are in the same block, a third image annotated with *orthodox icons with Jesus* is in a second block and another image tagged with *Manhattan* (concept which is loosely related to the query via *orthodox*) is in a third block. This is a kind of fusion since the order of the blocks keeps the $\mathcal{R}_t$ ranking but, within each block, we use the content-based ranking (cf. figure 3).

5 Experimental validation

Our method has been evaluated in the WikipediaMM task of the ImageCLEF 2009 campaign [4]. In these experiments, we consider each query as a concept.

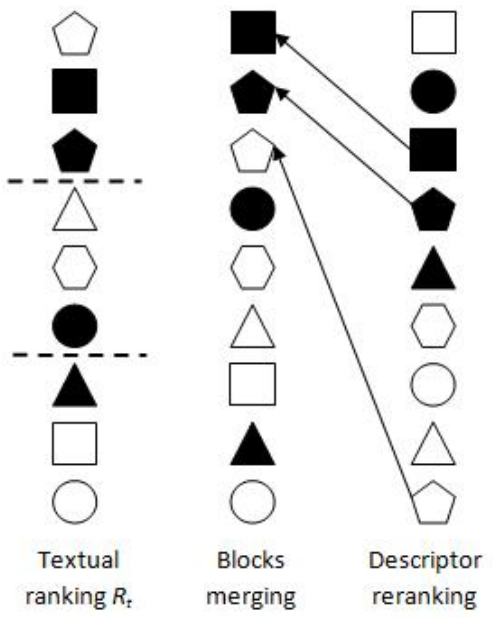

Figure 3: Scheme of the Block merging process

5.1 Experiments

For the visual prototypes creation, we query each query's heading using both Google and Yahoo! search engines to benefit from the difference between the index and improve our prototype variety. In our case, $N_q = 100$: only the top 50 pictures of each search engine results are selected to illustrate the concept. After processing the raw set, the new positive set contains $N_{pos} = 50$ images.

For practical reasons, all the concepts share the same $\mathcal{R}_{neg}$ negative set even if it should be ideally redefined for each query. Moreover, it is not a trivial task to build a coherent noisy class without relation with the concept, thus, in practice, this set contains concepts we consider as generic. We fixed N_{neg} to 300.

To compute the visual coherence score, N_{neigh} and N_{sum} are both fixed to 10 : the diversity of a concept can make two pictures semantically relevant but very far from each other according to a descriptor. The descriptors used are the color and texture based Local Edge Pattern (LEP, derived from [1]) and classical bags of features [6]. Hence, for each query, we obtain three visual prototypes : LEP visual prototype, Bag of feature visual prototype and the Early-fusion visual prototype which results from the mix of the two previous. Finally we get four content-based rerankings : the texture LEP reranking, the Bag of features reranking, the Late-fusion reranking and the Early-fusion reranking.

For the Window merging method, we use a window of $S_{window} = 10$ elements. For the Blocks merging method, we divide $\mathcal{R}_t$ according to textual score.

5.2 Results

The results of our methods are reported in table 1. Note that we highlight some mistakes in the generation of the runs using the window merging approach bringing about differences with the results providing with the official runs (see footnotes). We report here the results of the correct runs, these runs do not outperform our best submissions but are still better than text only baseline.

Run	Reranking procedure	MAP	P@10	P@20
cealateblock	Blocks merging($\mathcal{R}_t$ textual ranking, Late-fusion reranking)	0.2051	0.3622	0.2744
ceaearlyblock	Blocks merging($\mathcal{R}_t$ textual ranking, Early-fusion reranking)	0.2046	0.3556	0.2833
ceaearlyblockres	Blocks merging($\mathcal{R}_t$ textual ranking, Late fusion reranking) 1000 images maximum/concept	0.2046	0.3556	0.2833
ceabofblock	Blocks merging($\mathcal{R}_t$ textual ranking, Bag of features reranking)	0.1975	0.3689	0.2789
ceatlepblock	Blocks merging($\mathcal{R}_t$ textual ranking, Texture LEP reranking)	0.1959	0.3467	0.2733
ceabofblockres	Blocks merging($\mathcal{R}_t$ textual ranking, Bag of features reranking) 1000 images maximum/concept	0.1946	0.3689	0.2789
ceatlepblockres	Blocks merging($\mathcal{R}_t$ textual ranking, Texture LEP reranking) 1000 images maximum/concept	0.1934	0.3467	0.2733
ceaearlywindow	Window merging($\mathcal{R}_t$ textual ranking, Early-fusion reranking)	0.1715	0.2778	0.2144[1]
cealatewindow	Window merging($\mathcal{R}_t$ textual ranking,Late-fusion reranking)	0.1693	0.2778	0.2167[2]
ceatlepwindow	Window merging($\mathcal{R}_t$ textual ranking, Texture LEP reranking)	0.1622	0.2689	0.2122[3]
ceatxt		0.1604	0.2333	0.2022
ceabofwindow	Window merging($\mathcal{R}_t$ textual ranking, Bag of features reranking)	0.1591	0.2800	0.2133[4]

Table 1: Results of the CEA LIST at ImageCLEF 2009 WikipediaMM Task. Notations: bof - bags of (visual) features; tlep - texture - color descriptor; block - block fusion; window- window fusion.

5.3 Discussion

Globally, these results lead to highlight the following points:

- The use of the content-based reranking (i.e the second step) always improves the results compared to the text-based baseline. However, we noticed during our preliminary experiments that the results can decrease when the reranking is global (i.e without window or block merging approaches). We expected such results because we retain up to 1000 results for each topic and the test collection contains around 150000 images. Consequently, a large part of the 1000 answers are not relevant and it is useful to exploit the textual ranking.
- The block merging is more efficient than the sliding window approach. The difference between block and window is significant (around 3 points in MAP). This proves that splitting the textual results according to the relatedness of each image to the query makes sense and
- Bag-of-feature (local descriptors) give slightly better results than texture LEP descriptor (global features). When looking at the P@10 scores, we notice that the bag of features runs return the best results.
- Fusion reranking significantly improves MAP results (around 1 point) in comparison to simple descriptor reranking. This finding confirms that when dealing with diversified topics, it is interesting to merge local and global descriptors in order to obtain better results.
- The late-fusion reranking gives better results than the early-fusion one but the difference is not significant.

6 Conclusion and future work

We presented methods for expanding textual queries and merging textual and visual information that are adapted for retrieving images in large datasets. Emphasis was put on designing techniques which can easily be scaled-up to larger image repositories. Textual results were somehow

[1]official result with the wrongly generated run was map=0.1182 P@10=0.2267 P@20=0.1889
[2]official result with the wrongly generated run was map=0.1178 P@10=0.2267 P@20=0.1900
[3]official result with the wrongly generated run was map=0.0944 P@10=0.1889 P@20=0.1544
[4]official result with the wrongly generated run was map=0.0921 P@10=0.1800 P@20=0.1633

disappointing and we are currently investigating why this happened and how they can be improved. One interesting work direction is to replace the refinement part of the concept ranking with techniques such as explicit semantic analysis [8] and it to the current performances of the system. We will also test the effects of the query expansion on larger datasets (such as the Web corpus) in order to compare it to standard search engine results. We are confident that results are likely to improve in terms of precision and diversity because the related concepts cover various aspects of a topic.

A second line of work concerns the visual processing in our system. We designed a k-NN based method for image reranking which is fast enough to be introduced in a search engine's pipeline given that the images are preindexed. We will explore the effects of introducing other content descriptors. Visual coherence was used at an image level but it is easy to compute a similar metric for topics and try to use it in order to determine automatically whether the visual reranking will be efficient (intuitively, it will work well for visually coherent queries). Visual coherence at a topic level can also be exploited in order to determine which descriptor to use for the reranking (it is probable for the best results to be obtained for the descriptor which provides the best separation between the positive and the negative sets).

Finally, it is interesting to explore how to adapt the size of the window (window merging) or the limits of the blocks (Block merging) to each list returned for each topic.

7 Aknowledgement

We thank the Direction Générale des Entreprises for funding us through the regional business cluster Systematic (project POPS) and Cap Digital (project Mediatic).

References

[1] Cheng, Ya-Chun and Chen, Shu-Yuan. Image classification using color, texture and regions. Image Vision Computing, 21(9):759–776, 2003.

[2] Adrian Popescu, Hervé Le Borgne, Pierre-Alain Moëllic, Conceptual Image Retrieval over the Wikipedia Corpus Working notes for the CLEF 2008 Workshop, Aarhus, Denmark, 17-19 September 2008.

[3] Adrian Popescu, Hervé Le Borgne, Pierre-Alain Moëllic, Conceptuel Image retrieval over a Large Scale Database. In Evaluating Systems for Multilingual and Multimodal Information Access, Proceedings of the 9th Workshop of the Cross-Language Evaluation Forum, Lecture Notes in Computer Science, vol. 5706, Springer 2009.

[4] Theodora Tsikrika and Jana Kludas. Overview of the wikipediaMM task at ImageCLEF 2009. CLEF workng notes 2009, Corfu, Greece, 2009.

[5] Theodora Tsikrika and Jana Kludas. Overview of the wikipediaMM task at ImageCLEF 2008. In Evaluating Systems for Multilingual and Multimodal Information Access, Proceedings of the 9th Workshop of the Cross-Language Evaluation Forum, Lecture Notes in Computer Science, vol. 5706, pp. 539-550, Springer 2009.

[6] J. Zhang, M. Marszalek, S. Lazebnik, and Cordelia Schmid. Local features and kernels for classifcation of texture and object categories: An in-depth study. Technical Report RR-5737, INRIA Rhône-Alpes, 2005.

[7] A. W. .M .Smeulders, M. Worring, S. Santini, A. Gupta & R. Jain Content-Based Image Retrieval at the End of the Early Years, IEEE Trans. on Patt. Anal. and Machine Intell., vol. 22, No. 12, pp. 467-477, 1997

[8] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using Wikipedia-based explicit semantic analysis, Proceedings of IJCAI, pp. 1606-1611, 2007.