Query Refinement and User Relevance Feedback for Contextualized Image Retrieval

K. Chandramouli[1], T. Kliegr[2], J. Nemrava[2], V. Svatek[2], E. Izquierdo[1]

[1] Multimedia and Vision Research Group (MMV),
Electronic Engineering Department,
Queen Mary University of London,
Mile End Road, E1 4NS, London, UK.
Email: {krishna.chandramouli, ebroul.izquierdo}@elec.qmul.ac.uk
[2] Department of Information and Knowledge Engineering,
Faculty of Informatics and Statistics, University of Economics in Prague
Winston Churchill Sq.4, 130 67 Prague, Czech Republic
Email: {tomas.kliegr, nemrava, Svatek}@vse.cz

Keywords: Query refinement, Wikipedia, K-Means, Particle Swarm Optimisation, Relevance Feedback

Abstract

The motivation of this paper is to increase the user perceived precision of results of Content Based Information Retrieval (CBIR) systems with Query Refinement (QR), Visual Analysis (VA) and Relevance Feedback (RF) algorithms. The proposed algorithms were implemented as modules into K-Space CBIR system. The QR module discovers hypernyms for the given query from a free text corpus (Wikipedia) and uses these hypernyms as refinements for the original query. Extracting hypernyms from Wikipedia makes it possible to apply query refinement to more queries than in related approaches that use static predefined thesaurus such as Wordnet. The VA Module uses the K-Means algorithm for clustering the images based on low-level features. The RF Module uses the preference information expressed by the user to build user profiles by applying SOM-based supervised classification, which is further optimized by a hybrid Particle Swarm Optimization (PSO) algorithm. The experiments evaluating the performance of QR and VA modules show promising results.

1 Introduction

With the advances in computer technologies and the advent of World Wide Web (WWW), there has been an explosion in the amount and complexity of digital data being generated, stored, transmitted, analysed and accessed. Much of this information is multimedia in nature, which includes digital images, videos, audio, graphics and text data. However, the queries submitted to information retrieval systems asking for these resources are often ambiguous. To partially overcome this problem, query expansion and query refinement techniques are commonly applied, with WordNet[1] being one of the most common lexical resource (e.g. [6]) used for this purpose. Since WordNet is not suitable for refining queries for named entities (consider query for "Hasek") due to its limited scope, we use a more comprehensive resource, the encyklopedia Wikipedia, as the basis for query refinement. For the example query "Hasek", it allows us to find multiple hypernyms including "goalkeeper" and "writer". If the context of the query is available, it is possible to choose the correct hypernym and perform query refinement. For football context, the result for the refined query "Hasek goalkeeper" is supposed to exhibit higher precision than the original query. This assumption is indeed supported by our experiments.

The context of image search is, however, one of the open research challenges for WWW resources based search engines (such as Google[2], Yahoo[3]). Capturing user query context is severely hindered by the fact that user preference varies in time. Hence, user profile needs to be automatically updated based on user interactions with the system.

Content Based Image Retrieval (CBIR) systems use the visual content extracted from the images to search large scale databases in effort to overcome this "Semantic Gap" between the query and web resources. To achieve this objective, we introduce here the *K-Space Image Retrieval System*, a CBIR system that lies on the crossroads of Natural Language Processing (NLP) and visual analysis. The K-space system integrates *Query Refinement Module* which contextualizes queries by applying NLP techniques, *Visual Analysis Module* that clusters images based on low-level features and *Relevance Feedback Module* which applies machine learning algorithms to create user profiles based on low-level image features and feedback provided by users.

The rest of the paper is organised as follows. Section 2 briefly introduces the K-Space Retrieval System. Query Refinement, Visual Analysis and Relevance Feedback modules are presented in sections 3, 4 and 5 respectively. Section 6 discusses the experimental results, conclusions and future work are presented in section 7.

[1] http://wordnet.princeton.edu

[2] http://images.google.co.uk/

[3] http://images.search.yahoo.com/images

2 K-Space Content Management and Retrieval System

The project behind the K-Space Content Management and Retrieval System [3] aims to design, implement, validate and trial a secure and efficient system for both delivery and access of multimedia content, and for the creation and querying of large distributed databases. The motivation is twofold: to enable seamless and secure provider-to-customer delivery of digital content components across heterogenous channels, and to ensure user-friendly retrieval in large databases when fast speed of access in a protected environment is demanded.

Figure 1: K-Space User Interface for text based Keyframe Retrieval

The key technology benefits brought by K-Space System are Video retrieval using self-embedded metadata, scalable techniques for efficient metadata generation, video annotation and querying and use of low-level descriptors to access high-level semantic video information. The metadata model is based on the syntactical structure description of a generic media type.

In this paper, we focus only on a subsystem of the K-Space Content Management and Retrieval system, namely K-Space Image Search Subsystem (KISS). The use case scenario for this is presented in the following subsection. A more detailed overview of its architecture is present in subsection 2.2.

2.1 Use Case Scenario

Robert as an NHL sports fan frequently searchers the web for images of his favourite stars. The online image search engines he uses do not consider the context in which the search query is provided. For example, for query ”Lundqvist” the online image search engine finds many famous individuals in multiple professions such as politician, comedian, artist, guitarist, etc. Robert often has to browse through the complete list of images and manually filter out the irrelevant ones, i.e. images of non NHL player. He prefers a system which can add context to his search queries and retrieve only the relevant images within his search context. Also, with respect to presenting the results, he would prefer to obtain an overview of the images which are collected from the web. This would save him time on browsing the whole set of images.

The KISS focuses exactly on the same problems Robert is facing. Namely, on adding contextual information to the query, building user profiles and clustering results based on the visual information contained within the image.

2.2 K-Space Image Search System Overview

The architecture of KISS is user-centered (the user enters the query). The query is further processed in the Query Refinement Module, which *contextualizes* the query by discovering possible contexts (hypernyms) for the query. In current implementation this step relies on Wikipedia as the training corpus. The contextualised query is presented to the online image search engine and the system receives a ranked collection of images. In order to present an overview of this collection, a set of visual descriptors is extracted from the images by the Visual Analysis Module. These low-level descriptors are used as input features for clustering algorithm, which splits images into clusters based on their visual similarity. The most representative image of each cluster is presented to the user. The user then has the option of providing feedback to the system in terms of expected or relevant images. User profile is created by the Relevance Feedback Module on the basis of the feedback provided by the user. Figure 2 presents the proposed system architecture.

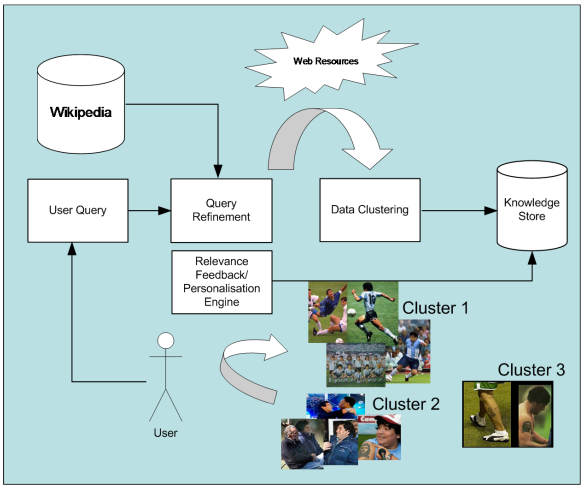

Figure 2: K-Space Image Search Engine Framework

3 Query Refinement

The Query Refinement Module essentially finds hypernyms for the original query and adds them the original query in

effort to obtain semantically purer results. Unlike most related approaches in this field, we neither rely on the availability of predefined taxonomy nor try to build one for later use. Instead, the goal is to discover hypernyms only for the current query based on a free text corpus. Query refinement in our approach thus depends on the availability of fast hypernym-discovery algorithm and high-coverage corpus. In this section, we give an outline of hypernym discovery algorithm that we developed that meets these requirements. First, we explain the choice of our corpus.

A gold standard resource for training and testing hypernym discovery algorithms is WordNet thesaurus (e.g. [5], [12]). As discussed in the introduction, WordNet has unfortunately too narrow coverage for the K-Space system. Actually, most publicly available well structured lexical resources including WordNet will fail when it is necessary to find hypernyms for domain specific search queries. We therefore opted for a loosely structured but large publicly available lexical resource, the Wikipedia encyklopedia. Due to its remarkable size, it is supposed to cover more domains, both in the number and detail, than other resources. Hypernym relations are not, however, directly available in Wikipedia and need to be discovered.

The discovery of hypernyms from Wikipedia is eased by its encyclopedic character, which brings three important properties: 1) small duplicity of articles: each meaning of given word/expression (subject) is mostly covered only by one article, 2) Wikipedia article conveys only one meaning of given subject and 3) there are strict guidelines on how articles are written.

Taking advantage of these properties, it is possible to define a set of lexico-syntactic patterns for extraction of hypernyms with high precision and recall. In our previous work [8], we have achieved the F-measure of 78% in this task.

Unlike the approach of [1] who combine web search and titles of Wikipedia article and hyperlinks contained in them for extraction of instances of arbitrary relations, we use first sections of Wikipedia articles as the sole source of information. The benefit of the proposed approach is smaller time complexity as its offline copy can be stored in local information retrieval system and only the first several sentences need to be processed by NLP tools.

The pseudocode of the hypernym discovery algorithm follows:

1. Find and fetch relevant articles on Wikipedia for the current query (articles are sorted according to relevancy)
2. For each article:
   (a) Delete any Wikipedia specific formatting, hidden text and information boxes
   (b) Preprocess the text
   (c) Apply lexico-syntactic patterns to extract hypernym from the first section of the article. If there are multiple candidate hypernyms, choose the first one.

A detailed discussion of the algorithm is presented in [8].

4 Visual Analysis

The objective of the Visual Analysis Module is two-fold. The rudimentary objective is to extract low-level features from images and use them for clustering. We used MPEG - 7 low-level features Colour Layout Descriptor (CLD) and Edge Histogram Descriptor (EHD) [10] for this purpose. The second objective is to use the extracted features as one of the inputs for the Relevance Feedback Module.

The clustering of images based on low-level visual features (CLD, EHD) is achieved using K-Means, which is computationally efficient for large datasets with both numeric and categorical attributes. For the traditional K-Means clustering algorithm, K samples are chosen at random from the whole sample space to approximate centroids of initial clusters. The K-Means clustering algorithm then iteratively updates the centres until no reassignment of patterns to new cluster centre occurs. In every step, each sample is allocated to its closest centre and cluster centres are reevaluated based on current cluster memberships. A detailed discussion on K-Means algorithm can be found in [13].

5 Relevance Feedback

The most natural way of obtaining user's subjective information and preferences is by using models that incorporate online learning from the user interactions with the search engine. The idea underpinning this model is to integrate a relevance feedback loop into the interaction between the system and the user. The concept of relevance feedback is based on the analysis of the user marking images as "relevant" and "irrelevant". The Relevance Feedback Module learns user's preferences by applying a machine learning algorithm. We have selected Self Organizing Map (SOM) for this task. The nodes of the SOM are trained with user input data, thus achieving supervised classification instead of conventional clustering. We improve the performance of the SOM with Particle Swarm Optimization (PSO).

In subsection 5.1 we give an overview of the relevance feedback loop, then we detail the techniques used within it by introducing SOMs in subsection 5.2, PSO in subsection 5.3) and then in subsection 5.4 we show how we these algorithms for improved relevance feedback.

5.1 Relevance Feedback Loop

KISS builds individual user profiles on incremental basis. Feedback is collected while the user browses the search results as the user can subjectively mark images as relevant or irrelevant. Until enough feedback is collected, the system operates in Clustering Mode. As the number of user interactions with the system increases beyond a certain threshold the system transfers to Classification Mode. For new search queries, the knowledge gathered from the user profile is used to perform supervised classification instead of clustering. In order to reflect the drift in the user's preferences over time, there is another threshold, which is set in terms of months, to e.g. 3, 6 months. When this threshold is reached, the system reverts from the Classification Mode back to the Clustering Mode while continuing to monitor the changes in user preferences.

5.2 Self Organizing Maps

Neural network based clustering and classification has been dominated by Self Organizing Maps (SOMs) [9] and Adaptive Resonance Theory (ART) [13]. In competitive neural networks, active neurons reinforce their neighbourhood within certain regions, while suppressing the activities of other neurons. This is called on-center/off-surround competition. The objective of SOM is to represent high-dimensional input patterns with prototype vectors that can be visualized in a usually two-dimensional lattice structure. Input patterns are fully connected to all neurons via adaptable weights. During the training process, neighbouring input patterns are projected into the lattice, corresponding to adjacent neurons. In the basic training algorithm are the prototype vectors trained according to the following equation:

$$m_d(t+1) \;=\; m_d(t) + g_{cd}(t)[x - m_d(t)]\,, \qquad (1)$$

where m_d is the weight of the neurons in the SOM network, $g_{cd}(t)$ is the neighbourhood function and d is the dimension of the input feature vector.

5.3 Particle Swarm Optimization

Particle Swarm Optimization (PSO) algorithm is one of the evolutionary computation techniques [4]. It was originally inspired by the social behaviour of a flock of birds [11] and further developed by Eberhart and Kennedy in 1995. In the PSO algorithm, the birds in a flock are considered to be "flying" through a problem space searching for a solution. The solution obtained by the particles is evaluated by a fitness function that provides quantitative value of the solution's utility. During each iteration, PSO changes the velocity (acceleration) of each particle toward its personal best ($pbest$) and global best ($gbest$). Acceleration is weighted by a random number, with separate random values being generated for acceleration toward $pbest$ and $gbest$. The commonly used versions of this algorithm are global and local PSO. The two versions differ in the definition of particle's neighbourhood, which is generally defined as topologically nearest particles to the particle on each side. In the local version of PSO, the neighbourhood of a particle includes a limited number of particles on its sides, while in the global version of PSO; it includes all the particles in the population.

5.4 Training SOM on Relevance Feedback data

Here we explain the setup and use of the SOM in our application. The size of the SOM network is predefined with the maximum number of feedback samples to be obtained from the user's interaction. During the initialization phase, the state of all nodes in the SOM network is set to '0'. During the training phase, the state of nodes will change to '1' (positive feedback) and '2' (negative feedback) depending on the user's decision.

The outcome is a partially (some nodes in state '0' representing untrained states) or completely (all nodes are in state '1' or '2') trained SOM classifier modeling user's preference. In the test phase, the trained network is presented with the test feature vectors to generate the ranked list. The images from the sorted ranked list are presented to the user. The user can again opt to provide feedback, which would initiate incremental training of the SOM as explained in subsection 5.1. To further improve the performance of the SOM classifier, the weights of neurons m_d from Equation 1 of SOM network are optimised with hybrid PSO algorithm. We call this algorithm "hybrid" because it combines both local and global version of the original algorithm as follows.

During each iteration, PSO changes the velocity of each particle toward its personal best ($pbest$) and global best ($gbest$). The velocity and position of the particles are governed by Equations 2 and 3.

$$v_{id} = v_{id} + c_1(pbest_i - x_{id}) + c_2(gbest_d - x_{id}) \qquad (2)$$

$$x_{id}(t) = x_{id}(t-1) + v_{id}(t-1) \qquad (3)$$

where v_{id} and x_{id} represent the velocity and position of individual particles i in each dimension d. The first summand of Equation 2 represents the velocity at previous time instant, which provides the necessary momentum for particles to roam across the search space. The second summand is known as cognitive component and represents the knowledge gained by individual particles. The third summand represents the social behaviour of particles. Social behaviour expresses the collaborative effect of the particles in finding the global optimal solution. The social component always pulls the particles

toward the global best particle found so far. The values of the parameters c_2 and c_1 determine the choice between the social and cognitive behaviour of the particles. After several experiments, these parameters were experimentally set to 0.68 and 0.32 respectively. We used PSO to train SOM in the following way: PSO receives the weight (m_d) from Equation 1 of the winning node of the SOM network as the input and performs the optimization. The optimized value of m_d is reassigned to the winner node of the SOM network. A detailed description of the PSO implementation can be found in [2]. The application of PSO for training SOM was found to perform well on other datasets (e.g. Corel) [7] and hence PSO is used in our framework. Evaluation of performance on relevance feedback data is left, however, for further work.

6 Experimental Results

We have performed two experiments when evaluating the performance of K-Space Image Retrieval System. The first experiment evaluates the contribution of the Query Refinement Module to image search results. The second experiment evaluates the precision of clustering of the Visual Analysis Module.

6.1 Query Refinement

The objective of this experiment is to verify that hypernym-based clustering can improve user experience with an image search engine. The experiment was carried out on the set of ten queries (surnames of famous NHL goalkeepers) presented in Figure 3. The use case scenario is that the user knows only the surname and wants to retrieve images related to the context of this goal keeper. In the experiment, the tasks in the experiments were conducted by three users and the results averaged.

The baseline was provided by the number of relevant and irrelevant images as subjectively annotated by the user in the results for the original query. This was compared with the results enhanced with the Query Refinement Module. On average nine top hypernyms were returned for each query. These hypernyms were used to produce clusters of images with hypernyms serving as their labels. For example, for query “Giguere”, the first four clusters were “goaltender”, “manager”, “swimmer” and “pioneer”. The user chose the cluster labeled “goaltender”, because it had the most apt label to the context of the query. The images for the cluster “goaltender” were retrieved from Yahoo Image API using the query “Giguere goaltender” and the user then marked images as either relevant or irrelevant. It can be observed from the experimental results that the percentage of retried relevant images on the total number of images is almost in all cases higher for the refined queries. Queries “Miller” and “Leclaire” were outliers, there were none or too few images marked as relevant in the baseline.

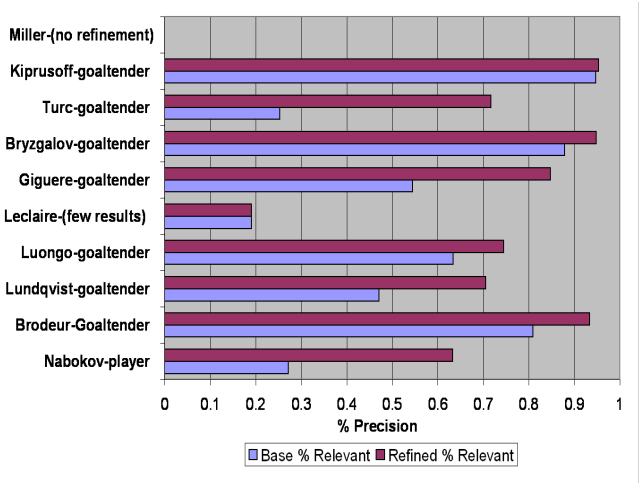

Figure 3: Experimental Results for Query Refinement

From the experimental results, we observed that refined queries provided consistently better precision with average relative improvement of 36However, because there is usually abundance of images matching the query when using web search, we believe that in most cases, the increase in precision outweights smaller recall. In any case, the user is given the option to inspect the original resultset.

6.2 Visual Analysis

The visual analysis includes feature extraction, clustering and user relevance feedback. The MPEG - 7 low-level visual features (CLD and EHD) are extracted from the images obtained from the web in this step. Following the feature extraction, images are clustered with K-Means algorithm. The objective of this step is to cluster the images according to visual similarity, so that the user can be presented with the most representative images from each cluster thus avoiding the user to manually browse all the images for collection. In our experiment, we asked two users to evaluate the precision of clustering of images retrieved for the original and refined query. In both cases, the number of clusters the users were presented with was experimentally set to 5.

The users were asked to annotate the presence of relevant images in each of the clusters with regard to the context of the query (NHL hockey players). The experimental results are presented in Table 1. It can be observed that the retrieval precision in general increases for refined queries, thus providing the user with more appropriate set of retrieved images. However for the query “Miller”, there were no relevant images retrieved thus the retrieval precision was 0% as also discussed in the previous experiment. Overall, the Visual Analysis Module improved the user perceived precision by 27.085% (refined vs original queries).

Query	Query Type	Retrieval Precision				
Miller	Original	0%	0%	0%	0%	0%
	Refined	46%	0%	0%	0%	0%
Kiprusoff	Original	80%	50%	93.33%	100%	89.47%
	Refined	100%	75%	100%	90.91%	100%
Turco	Original	33.33%	0%	36.36%	0%	12.5%
	Refined	50%	100%	80%	72.72%	33.33%
Bryzgalov	Original	60%	77.77%	80%	88%	68.75%
	Refined	100%	100%	100%	80%	100%
Giguere	Original	33.33%	14.28%	58.33%	34.78%	66.67%
	Refined	75%	100%	100%	58.33%	100%
Leclaire	Original	0%	14.28%	0%	6.67%	23.411%
	Refined	100%	0%	0%	100%	0%
Luongo	Original	40%	66.67%	60%	55%	59.1%
	Refined	57.14%	50%	100%	50%	100%
Lundqvist	Original	28.57%	0%	100%	62.5%	12.5%
	Refined	100%	50%	100%	100%	100%
Brodeur	Original	53.84%	16.67%	93.75%	75.86%	83.33%
	Refined	100%	20%	91.67%	87.6%	100%
Nabakov	Original	0%	0%	25%	60%	40%
	Refined	0%	0%	57.14%	83.33%	0%
Cluster ID		1	2	3	4	5

Table 1: Clustering Precision subjectively annotated by humans

7 Conclusion and Future Work

In this paper we presented an image retrieval system using enhanced query processing and relevance feedback. Queries are refined with hypernyms extracted from Wikipedia in order to contextualise the user query for image search. Experiment 1 showed that this improves the precision of search results by 36% as images with undesirable meaning are removed. It should be noted that this increase in precision comes at the expense of number of images retrieved. The Visual Analysis Module has shown to contribute towards improving the user perceived precision by 27%. The Relevance Feedback Module is used to obtain the user preferences for the set of retrieved images. Thus the user is able to build a profile of relevant images, which can be used for further processing of retrieved results. The performance of the Relevance Feedback Module will be evaluated in the future work.

Acknowledgements

The research work leading to this paper has been partially supported by the European Commission under the IST research network of excellence K-SPACE of the 6th Framework programme. The authors also acknowledge the efforts of fellow researchers from the Multimedia and Vision Group for their invaluable contribution in developing the system and also the users participated in the study.

References

[1] S. Blohm, P. Cimiano, and E. Stemle. Harvesting relations from the web - quantifiying the impact of filtering functions. In *AAAI*, pages 1316–1321, 2007.

[2] K. Chandramouli and E. Izquierdo. Image classification using self organising feature maps and particle swarm optimisation. In *Proc. 7th Int'lWorkshop on Image Analysis for Multimedia Interactive Services (WIAMIS'06)*, pages 313–316, 2006.

[3] M. Grzegorzek E. Izquierdo, K. Chandramouli and T. Piatrik. K-space content management and retrieval system. In *Proc. 14th International Conference on Image Analysis and Processing*, 2007.

[4] R. C. Eberhart and Y. Shi. Tracking and optimizing dynamic systems with particle swarms. *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, 1, 2001.

[5] Z. Gong, C. W. Cheang, and U. L. Hou. Web query expansion by wordnet. In *In Proceedings of DEXA 2005. LNCS 3588*, pages 166–175, 2005.

[6] L. Hollink, G. Schreiber, and B. Wielinga. Patterns of semantic relations to improve image content search. In *Web Semant.*, pages 195–203, 2007.

[7] E. Izquierdo K. Chandramouli. A study on relevance feedback using particle swarm optimization. In *Submitted to International Conference on Image Processing, ICIP'08*, 2008.

[8] T. Kliegr, K. Chandramouli, J. Nemrava, V. Svatek, and E. Izquierdo. Refining image search queries with hypernyms extracted from wikipedia. In *Submitted to, Workshop on Wikipedia, part of International Conference on Artificial Intelligence*, 2008.

[9] T. Kohonen. The self organizing map. *Proceedings of IEEE*, 78(4):1464–1480, September 1990.

[10] B. S. Manjunath, J-R. Ohm, V. V. Vinod, and A. Yamada. Color and texture descriptors. *IEEE Trans. Circuits and Systems for Video Technology, Special Issue on MPEG - 7*, 11(6):703–715, June 2001.

[11] C.W. Reynolds. Flocks, herds and schools: a distributed behavioural model. In *Computer Graphics*, pages 25–34, 1987.

[12] R. Snow, D. Jurafsky, and Y. Ng. Semantic taxonomy induction from heterogenous evidence. In *ACL., 2006*, 2006.

[13] R. Xu and D. Wunch II. Survey of clustering algorithms. *IEEE Trans. Neural Network*, 6(3):645–678, May 2005.