Context-Based Clustering of Image Search Results

Hongqi Wang[1,2], Olana Missura[1], Thomas Gärtner[1], and Stefan Wrobel[1,2]

[1] Fraunhofer Institute Intelligent Analysis and Information Systems IAIS, Schloss Birlinghoven, D-53754 Sankt Augustin, Germany
firstname.lastname@iais.fraunhofer.de
[2] Department of Computer Science III, Rheinische Friedrich-Wilhelms-Universitt Bonn, Römerstrasse 164, D-53117 Bonn, Germany
firstname.lastname@iai.uni-bonn.de

Abstract. In this work we propose to cluster image search results based on the textual contents of the referring webpages. The natural ambiguity and context-dependence of human languages lead to problems that plague modern image search engines: A user formulating a query usually has in mind just one topic, while the results produced to satisfy this query may (and usually do) belong to the different topics. Therefore, only part of the search results are relevant for a user. One of the possible ways to improve the user's experience is to cluster the results according to the topics they belong to and present the clustered results to the user. As opposed to the clustering based on visual features, an approach utilising the text information in the webpages containing the image is less computationally intensive and provides the resulting clusters with semantically meaningful names.

Key words: image clustering, machine learning, image search

1 Introduction

The information explosion brings a rapidly increasing amount of published information and makes a huge number of images available on the internet. Although search engines have made retrieval and managing of large amount of information from the internet much easier, the results from image search engines are not always satisfactory. Due to the ambiguity and context-dependency of human languages the same word can relate to wildly different things (consider, for example, that "jaguar" could refer to an animal, as well as to a car). As a result, for the same query an image search engine can return images from several different categories. In general users are interested in one particular category. Search engines such as Google [5], Yahoo! [17], and Picsearch [12] return a long list of image results, which users have to navigate by themselves by examining titles and thumbnails of these images to find the relevant results. It is a time-consuming and frustration-inducing task, especially when a large number of different topics is presented. A natural idea is to cluster the image search results returned by a search engine according to the topics they belong to.

In recent years several algorithms were developed based on Content-Based Image Retrieval (CBIR) [10, 9, 16, 15, 4, 2]. Unfortunately, all of them suffer from the following problems: First, they use high dimensional visual features that are too computationally intensive to be practical for web applications. Second, the generated clusters do

not have semantic names. [6] proposed IGroup as an efficient and effective algorithm. IGroup firstly builds different clusters with unique names by clustering text search results from Google Search and snippets from Picsearch using Q-grams algorithm [3]. The cluster names are then used to perform the image search on Picsearch. IGroup claims to have three unique features. First, the most representative image groups can be found with meaningful names. Second, all resulting images are taken into account in the clustering process instead of the small part. And third, this algorithm is efficient enough to be practical. In spite of these features, this algorithm, however, has some problems which have not been stated by the authors. In order to generate the cluster names, the system has to be well trained beforehand with some training data, which is improper for a real ISRC system, because for real systems query text is usually unknown and random.

In contrast to the approaches described above we propose a way to cluster the images based on the textual information contained in their referring webpages. The assumption is that given a webpage, its textual and visual content are related to each other. Furthermore, we assume that the distance within the webpage indicates the degree of relevance between the image and the text. There are two significant advantages of using clustering based on text as opposed to visual features. First, it is less computationally intensive, which is important in the context of web search, where the reaction times are on average less than a second. Second, using text features gives us an ability to construct semantically meaningful names for the clusters, which simplifies the navigation for the users.

In this paper we introduce TeBIC (Text Based Image Clustering) which clusters image search results into different category groups. We proceed as follows: In Section 2 the architecture of the TeBIC's components is described. After that the experimental results are presented in Section 3. The work is summarised and possible directions for the future work are outlined in Section 4.

2 Component Description

The images to cluster together with the information about their referring webpages are fetched from the Yahoo Image Search Engine [17]. Prior to clustering, TeBIC utilises a language filter to discard all the websites that are not written in a specified language. In our work English was used as a primary language, but any other language can be chosen due to the language independence property of the Q-gram algorithm. Then the data is extracted from the webpages using a content extractor and resulting feature vectors are clustered. After the clusters are generated, the cluster labeler assigns semantic names to the clusters according to the common topic of the images in the cluster. For each component we investigated several options, which are described below.

2.1 The Language Filter

Images of similar topics might be contained in the webpages written in different languages. A clustering algorithm using textual features is likely to cluster different languages into different clusters. To exclude this possibility TeBIC uses a language filter.

The Q-gram-based text categorisation algorithm [3] is used to filter out all webpages that are not written in English. All the words of a webpage are scanned to produce a list of Q-grams. The list is sorted in a descending order of their count (Q-gram that has the largest count comes first) and stored as the document profile of the webpage. The webpage is assigned to a language such that the distance between their profiles is minimal. (For a detailed description of the algorithm and the metrics used see [3].)

2.2 The Content Extractor

Text-based The text-based extractor takes into account only pure text information of a webpage, excluding scripts, comments and tag attributes. The resulting text document is represented by its the bag of words. The stop words are removed and the remaining words are stemmed into terms by Porter Stemmer [13]. After that the terms are weighted using the tf-idf approach with a sublinear tf scaling:

$$weight = \begin{cases} (1 + \log tf) \times \log \frac{N}{df_t} & \text{if } tf > 0; \\ 0 & \text{if } tf = 0; \end{cases} \tag{1}$$

where N is the total number of the webpages, tf is the term frequency and df_t is the inverse document frequency of each term.

DOM-based As opposed to plain text documents, webpages have a structural organisation, which is embodied in a Document Object Model (DOM) tree. Recall that we assumed that the text located closer to the image is more relevant to the topic of the image than the text located further away. The DOM tree allows us to utilise this information by calculating the distance between the image nodes and the text nodes in the following way:

- The distance of the target image node to itself is $zero$.
- In a subtree which contains the target image node, the distance of any child node, except the target image node itself and its children, is the difference of depth between the target image node and the least common ancestor of the child node and the target image node.

The weight w_i of a text node n_i is calculated according to the idea that the closest nodes to the image are the most important ones and with the increasing distance their importance is rapidly falling:

$$w_i = \frac{1}{\sum_{i=1}^{N} e^{-\frac{d_i^2}{3}}} e^{-\frac{d_i^2}{3}}, \tag{2}$$

where d_i is the distance between the ith text node and the target image node and the constant 3 was determined empirically. The frequency of the term t is then scaled with the weight of its text node to calculate the final term weight:

$$wf_t = \sum_{i=1}^{N} f_t|_{n_i} \times w_i, \tag{3}$$

where w_i is the weight of the ith text node, $f_t|_{n_i}$ is the term frequency in the text node n_i.

Link-based Another difference between a webpage and a text document is that a webpage contains hyperlinks to other webpages, which may provide additional useful information. The link-based extractor searches for all the non-nepotistic (i.e. leading to other domains) hyperlinks in all webpages returned by the search engine. Each webpage is then represented by a vector V of links according to the occurrence of the hyperlinks.

External-Page and DOM-based Not all images have referring webpages with enough textual information (e.g. photo galleries). To overcome this drawback we designed an external-page and DOM-based extractor. It augments the word vector constructed by the DOM-based extractor with the terms mined from the webpages that the non-nepotistic hyperlinks lead to. The new terms are weighted with the the minimal weight value presented in the original word vector.

2.3 Cluster Analyser

K-means The K-means algorithm [14] clusters n objects into k partitions, $k < n$, by minimising intra-cluster distances.

Yet another K-means The Yet another K-means (YAK) algorithm was designed by the authors to overcome a drawback [1] of the K-means algorithm, namely the need for the parameter specifying the number of clusters upfront. Instead of a pre-defined number of clusters, YAK requires a maximum possible number of clusters k. YAK is a soft-clustering algorithm and its resulting clusters may overlap. It proceeds as follows:

1. One of the data points is selected randomly and assigned to the initial singleton cluster.
2. A data point x is assigned to each cluster c_i such that the similarity between c_i and x is above the similarity threshold s. Note that in this step the point x can be assigned to more than one cluster. If no such cluster exists, x forms a new singleton cluster. In case that the maximum cluster number k is reached, x is assigned to c_i with the maximum similarity value. The centroids of clusters are recalculated and the process is repeated until clusters no longer change.
3. Clusters which share most of their elements are merged according to a merge threshold m. After the merging process the singleton clusters are discarded.

One may say that we replaced one parameter (number of clusters) with two new ones (similarity and merge thresholds). For the number of clusters there is no way to know in advance how many topics the resulting images belong to. The thresholds, on the other hand, have universal values that can be established empirically.

Non-negative matrix factorisation (NMF) The goal of the NMF [7] is to factorise the feature space (represented by a set of feature vectors) into a k-dimensional semantic space with each dimension corresponding to a particular topic and n-dimensional weight space. Each topic represents a cluster. In the semantic space each item, in our case each referring webpage, is represented by a linear combination of the k topics. For the details about the NMF algorithm see [7].

2.4 Cluster Labeler

Information Gain (IG) The IG [18] approach calculates the importance of each term to each cluster as follows:

$$IG(T|C) = \sum_{e_c \in \{1,0\}} \sum_{e_t \in \{1,0\}} P(T = e_t, C = e_c) \log \frac{P(T = e_t, C = e_c)}{P(T = e_t)P(C = e_c)} \quad (4)$$

where T has values $e_t = 1$ when term t_i is in a webpage in the webpage pool and $e_t = 0$ when term t_i is not. Similarly C has values $e_c = 1$ when the webpage is in the cluster ${}_kc$ and $e_c = 0$ when the webpage is not. The terms are sorted in a descending order according to their IG values for each cluster and the top ten are used as the cluster's names.

χ^2 Test The χ^2-test [18] approach tests the independence between each cluster and each term:

$$\chi^2(T, C) = \sum_{e_t \in 1,0} \sum_{e_c \in 1,0} \frac{(O_{e_t e_c} - E_{e_t e_c})^2}{E_{e_t e_c}} \quad (5)$$

where e_t and e_c have the same definition as in the IG approach. $O_{e_t e_c}$ and $E_{e_t e_c}$ are the *observed frequency* and the *expected frequency* that take the values of e_t and e_c.

Word Frequency (WF) The WF approach sorts the terms in a descending order according to their weight in a cluster and uses the top ten as the cluster's names.

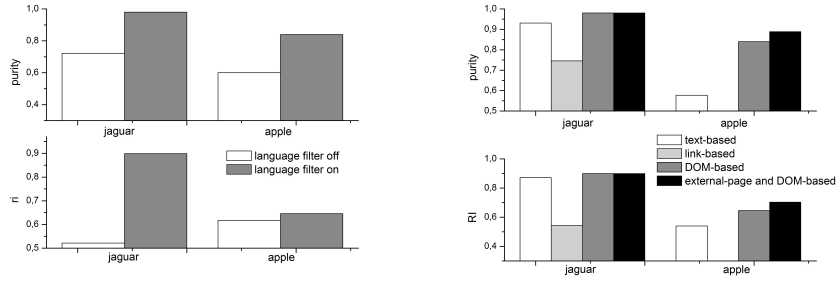
3 Experimental Results

The goal of the evaluation was to determine the best choice of the TeBIC's components. To this purpose we tested its performance with the language filter and without, when utilising each of the described content extractors and each of the clustering algorithms. Furthermore, it was interesting to determine which of the proposed labeling methods provides semantically better cluster name. The quality of the labels was evaluated using human judgement. Two metrics were used in the evaluation, purity and Rand Index [8]:

$$purity(K, C) = \frac{1}{N} \sum_{c_i \in C} \max_j |k_j \cap c_i| \quad (6)$$

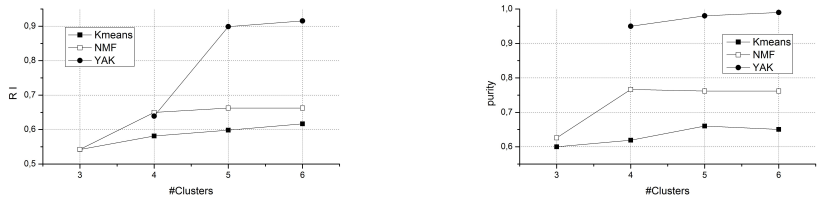
$$RI = \frac{tp + tn}{\binom{N}{2}}, \quad (7)$$

where $K = \{k_1, \cdots, k_m\}$ is the set of topics, $C = \{c_1, \cdots, c_n\}$ is the set of clusters, $|k_j \cap c_i|$ is the number of images from topic k_j that are clustered in cluster c_i, tp is the number of true positives, tn is the number of true negatives, and N is the total number of images.



(a) With and without language filter. (b) With different content extractors.

Fig. 1. TeBIC's performance.



(a) (b)

Fig. 2. Purity and RI over different number of clusters generated by different clustering algorithms. #Clusters is the actual cluster number used by the corresponding algorithm. The upper bound for YAK was set to 10.

We evaluated the effect that different components have on the performance of TeBIC on two query terms, "jaguar" and "apple", chosen for their ambiguity. Figure 2(a) shows that both purity and RI increase when the language filter is on. Figure 2(b) shows the purity and RI values for each of the proposed content extractors. The two DOM-based methods outperform the text-based extractor according to both metrics. From Figure **??** we can see that the YAK cluster analyser outperforms the other two approaches probably due to it being a soft-clustering method and removing the singletons or small clusters in the end. The comparison of the labelers is presented in Table 1. The German terms used in the labels occur in the <META> tags of some of the referring webpages even when the main text is written in English. Note that according to [18] IG and CHI

scores of a term are strongly correlated. The examination of the suggested cluster names reveals that a simpler method of counting the word frequencies generates better names than the other two. It is possible that this is due to a small number of clusters produced in our experiments. [8] showed that the performance of the frequency-based selection method falls quickly when the number of clusters is larger than 10.

Based on the collected data we conclude that the best configuration from the suggested options is TeBIC consisting of the language filter, DOM-based content extractor, soft clustering algorithm YAK, and word frequency-based labeler.

	IG	CHI test	Word Frequency	Real Topic
1	**mieten leihen** hochzeitswagen verleih **werkstatt** fahrzeuge reparaturen **werkst** jaguarverkauf klassische	**mieten leihen** hochzeitswagen verleih **werkstatt** fahrzeuge reparaturen **werkst** jaguarverkauf klassische	type oldtimer **ausfahrten** classical **club veranstaltungen** ersatzteile selber klassik fahren	jaguar club
2	**south animal mother america live** information space **rainforest** weighting prey	**south animal mother america live** information space **rainforest** weighting prey	**panthers** frazetta **animal** mayan **cats** size black walks **leopard** fear	jaguar cat
3	lancia facel tomaso **allsportauto** lada alphine marcos **oldsmobile** tatra caterham	lancia facel tomaso **allsportauto** lada alphine marcos **oldsmobile** tatra caterham	andros mondial louis photos **allsportauto bagatelle retromobile sportive** masini trophe	jaguar car
4	nutz silly eclectech flappy bytemark cash tube shirts hello	nutz silly eclectech flappy bytemark cash tube shirts hello	**specs** technically **specific xkr** parks image arden picture gray exotic	mainly cars, mixed with cats

Table 1. Automatically computed cluster labels for the query "jaguar".

4 Conclusion and Future Work

In this paper we proposed an approach to cluster image search results based on the textual information as a way to overcome the problems of visual features based algorithms, namely their high computational costs and lack of semantic names for the generated clusters. The preliminary results demonstrate the soundness of the idea that the text in the referring webpages provides enough information for the clustering of the images. However, further experiments are required to compare the performance of TeBIC with other approaches (e.g. IGroup [6]). In future work we also intend to conduct user studies to answer the question, whether clustering of image search results indeed improves the user experience.

Acknowledgments

We would like to thank the anonymous reviewers for their helpful and insightful comments.

References

1. Berkhim, P: Survey of clustering data mining techniques, Tech. rep., Accrue Software, San Jose, CA (2002)
2. Cai, D., X. He, Z. Li, W.-Y. Ma, and J.-R. Wen: Hierarchical clustering of WWW image search results using visual, textual and link information, in MULTIMEDIA 04: Proceedings of the 12th annual ACM international conference on Multimedia, New York, NY, USA: ACM, 952959 (2004)
3. Cavnar, W. B. and J. M. Trenkle: N-gram-based text categorization, in In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, pp. 161–175 (1994)
4. Gao, B., T.-Y. Liu, T. Qin, X. Zheng, Q.-S. Cheng, and W.-Y. Ma: Web image clustering by consistent utilization of visual features and surrounding texts, in MULTIMEDIA 05: Proceedings of the 13th annual ACM international conference on Multimedia, New York, NY, USA: ACM, 112121 (2005)
5. Google Image Search, `http://images.google.com`
6. Jing, F., C. Wang: IGroup: web image search results clustering, in MULTIMEDIA '06. New York, NY, USA: ACM, 377-384 (2006)
7. Lee D. D., H. S. Seung: Algorithm for non-negative matrix factorization, MIT Press, 556-562
8. Manning, C. D., P. Raghvavan, H. Schütze: An Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
9. Liu, X. and W. B. Croft: Cluster-based retrieval using language models, in SIGIR 04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, New York, NY, USA: ACM, 186193 (2004)
10. Luo, B., X. Wang, and X. Tang: World Wide Web Based Image Search Engine Using Text and Image Content Features, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, ed. by S. Santini and R. Schettini, vol. 5018 of Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, 123130 (2003)
11. Nielson Online, `http://www.nielsen-online.com`
12. Picsearch, `http://www.picsearch.com`
13. Porter, M. F.: An algorithm for suffix stripping, Program, 313–316
14. Steinhaus, H.: Sur la division des corp materiels en parties, Bull. Acad. Polon. Sci, 1, 801-804
15. Wang, Y. and M. Kitsuregawa: Use Link-Based Clustering to Improve Web Search Results, in WISE 01: Proceedings of the Second International Conference on Web Information Systems Engineering (WISE01) Volume 1, Washington, DC, USA, 115 (2001)
16. Wang, X.-J., W.-Y. Ma, Q.-C. He, and X. Li: Grouping web image search result, in MULTIMEDIA 04: Proceedings of the 12th annual ACM international conference on Multimedia, New York, NY, USA: ACM, 436439 (2004)
17. Yahoo! Image Search, `http://images.search.yahoo.com`
18. Yang, Y. and J. O. Pedersen: A Comparative Study on Feature Selection in Text Categorization, in ICML 97: Proceedings of the Fourteenth International Conference on Machine Learning, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 412420 (1997)