

Web Image Retrieval Re-Ranking with Relevance Model

Wei-Hao Lin, Rong Jin, Alexander Hauptmann
Language Technologies Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA, 15213 U.S.A
{whlin,rong,alex}@cs.cmu.edu

Abstract

Web image retrieval is a challenging task that requires efforts from image processing, link structure analysis, and web text retrieval. Since content-based image retrieval is still considered very difficult, most current large-scale web image search engines exploit text and link structure to “understand” the content of the web images. However, local text information, such as caption, filenames and adjacent text, is not always reliable and informative. Therefore, global information should be taken into account when a web image retrieval system makes relevance judgment. In this paper, we propose a re-ranking method to improve web image retrieval by reordering the images retrieved from an image search engine. The re-ranking process is based on a relevance model, which is a probabilistic model that evaluates the relevance of the HTML document linking to the image, and assigns a probability of relevance. The experiment results showed that the re-ranked image retrieval achieved better performance than original web image retrieval, suggesting the effectiveness of the re-ranking method. The relevance model is learned from the Internet without preparing any training data and independent of the underlying algorithm of the image search engines. The re-ranking process should be applicable to any image search engines with little effort.

1. Introduction

As World-Wide Web grows in an exploding rate, search engines become indispensable tools for any users who look for information on the Internet, and web image search is no exception. Web image retrieval has been explored and developed by academic researchers as well as commercial companies, including academic prototypes (e.g. VisualSEEK [20]), additional search dimension of existing

web search engines (e.g. Google Image Search [10], AltaVista Image [1], specialized web image search engines (e.g. Ditto [8], PicSearch [18]), and web interfaces to commercial image providers (e.g. Getty Images [9], Corbis [6]).

Although capability and coverage vary from system to system, we can categorize the web image search engines into three flavors in terms of how images are indexed. The first one is text-based index. The representation of the image includes filename, caption, surrounding text, and text in the HTML document that displays the image. The second one is image-based index. The image is represented in visual features such as color, texture, and shape. The third one is hybrid of text and image index. However, text-based index seems to be the prevailing choice now if anyone plans to build a large-scale web image retrieval system. Possible reasons include: text input interface allows users to express their information need more easily than image interface, (asking users to provide a sample image or drawing a scratch is seldom feasible), image understanding is still an open research problem, and image-based index are usually of very high dimensionality.

Most web image search engines provide a text input interface (like HTML tag `<INPUT>`) that users can type keywords as a query. The query is then processed and matched against the indexed web images, and a list of candidate images are ranked in the order of relevance before results are returned to users, as illustrated in Figure 1.

However, textual representation of an image is often ambiguous and non-informative of the actual image content. Filenames may be misleading, adjacent text is difficult to define, and a word may contain multiple senses. All these factors confound the web image retrieval system. More context cues should be taken into consideration when the web image retrieval systems managed to disambiguate and rank images.

One piece of information in the HTML documents that can help make relevance judgment is link structure. Sophisticated algorithms such as PageRank [3], “Hub and Au-

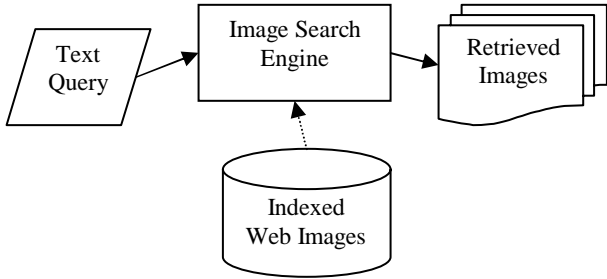


Figure 1. A high-level overview of web image search engine

thorities” [14] rank documents by analyzing the link structure between documents. A document is more important if it links many “good” pages, and many “good” pages link it. Similar ideas have been applied to web image retrieval (e.g. PicASHOW [16]), and images are ranked by considering the web page is an image container or a hub. However, for outsiders to make use of link structure, the index information of the web image search engine must be publicly accessible, which is unlikely and sometimes impossible.

In the paper, we propose a re-ranking process to reorder the retrieved images. Instead of accepting the results from a web image search engine, the image rank list as well as associated HTML documents are fed to a re-ranking process. The re-ranking process analyzes the text of the HTML document associated with the images to disambiguate document/image relevance using a relevance model. The relevance model is built automatically through a web text search engine. The re-ranking process (above the dashed line) is illustrated in Figure 2.

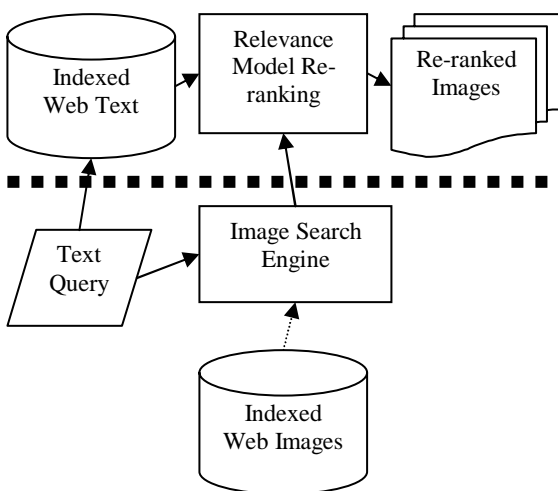


Figure 2. An overview of web image retrieval re-ranking

The basic idea of re-ranking is that the text part of HTML documents (i.e., after removal of all HTML tags in the HTML documents) should be relevant to the query if the image displayed in the document is relevant. For example, when a user input a text query “Statue of Liberty” to a web image search engine, we expect the web pages with images relevant to query is more likely to be history or travel information for “Statue of Liberty”, but less likely to be pages describing a ship happening to be named after “Statue of Liberty”.

We describes the relevance model, the key component in the re-ranking process, in Section 2. Experiments are conducted to test the re-ranking idea in Section 3. The connection between relevance model re-ranking to the Information Retrieval techniques are discussed in Section 4. Finally we conclude the paper, and present some directions of future works.

2. Relevance Model

Let us formulate the web image retrieval re-ranking problem in a more formal way. For each image I in the rank list returned from a web image search engine, there is one associated HTML document D displaying the image, that is, the HTML document D contains an `` tag with `src` attribute pointing to the image I . Since both image understanding and local text information are exploited by the image search engine, we wonder if we can re-rank the image list using global information, i.e. text in the HTML document, to improve the performance. In other words, can we estimate the probability that the image is relevant given text of the document D , i.e. $\Pr(R|D)$? This kind of approach has been explored and called Probability-Based Information Retrieval [2].

By Bayes’ Theorem, the probability can be rewritten as follows,

$$\Pr(R|D) = \frac{\Pr(D|R) \Pr(R)}{\Pr(D)} \quad (1)$$

Since $\Pr(D)$ is equal for all documents and assume every document is equally possible, only the relevance model $\Pr(D|R)$ is needed to estimate if we want to know the relevance of the document, which consequently implies the relevance of the image within.

Suppose the document D is consisted of words $\{w_1, w_2, \dots, w_n\}$. By making the common word independence assumption [21],

$$\Pr(D|R) \approx \prod_{i=1}^n \Pr(w_i|R) \quad (2)$$

$\Pr(w|R)$ can be estimated if training data are available, i.e. a collection of web pages that are labeled as relevant

to the query. However, we cannot afford to collect training data for all possible queries because the number of queries to image search engines everyday is huge.

2.1. Approximate Relevance Model

A method, proposed by Lavrenko and Croft [15], offers a solution to approximate the relevance model without preparing any training data. Instead of collecting relevant web pages, we can treat query Q as a short version of relevant document sampling from relevant documents,

$$\Pr(w|R) \approx \Pr(w|Q) \quad (3)$$

Suppose the query Q contains k words $\{q_1, q_2, \dots, q_k\}$. Expand the conditional probability in Equation 3,

$$\Pr(w|Q) = \frac{\Pr(w, q_1, q_2, \dots, q_k)}{\Pr(q_1, q_2, \dots, q_k)} \quad (4)$$

Then the problem is reduced to estimate the probability that word w occurs with query Q , i.e. $\Pr(w, q_1, q_2, \dots, q_k)$. First we expand $\Pr(w, q_1, q_2, \dots, q_k)$ using chain rule,

$$\Pr(w, q_1, q_2, \dots, q_k) = \Pr(w) \prod_{i=1}^k \Pr(q_i|w, q_{i-1}, \dots, q_1) \quad (5)$$

If we further make the assumption that query word q is independent given word w , Equation 5 becomes

$$\Pr(w, q_1, q_2, \dots, q_k) \approx \Pr(w) \prod_{i=1}^k \Pr(q_i|w) \quad (6)$$

We sum over all possible unigram language models M in the unigram universe Ξ to estimate the probability $\Pr(q|w)$, as shown in Equation 7. Unigram language model is designed to assign a probability of every single word. Words that appear often will be assigned higher probabilities. A document will provide a unigram language model to help us estimate the co-occurrence probability of w and q .

$$\Pr(w, q_1, q_2, \dots, q_k) = \Pr(w) \prod_{i=1}^k \sum_{M \in \Xi} \Pr(q_i, M|w) \quad (7)$$

In practice, we are unable to sum over all possible unigram models in Equation 7, and usually we only consider a subset. In this paper, we fix the unigram models to top-ranked p documents returned from a text web search engine given a query Q .

If we further assume query word q is independent of word w given the model M , Equation 7 can be approximated as follows,

$$\Pr(w, q_1, q_2, \dots, q_k) \approx \Pr(w) \prod_{i=1}^k \sum_{j=1}^p \Pr(M_j|w) \Pr(q_i|M_j) \quad (8)$$

The approximation modeled in Equation 8 can be regarded as the following generative process: we pick up a word w according to $\Pr(w)$, then select models by conditioning on the word w , i.e. $\Pr(M|w)$, and finally select a query word q according to $\Pr(q|M)$.

There are still some missing pieces before we can actually compute the final goal $\Pr(D|R)$. $\Pr(q_1, q_2, \dots, q_k)$ in Equation 4 can be calculated by summing over all words in the vocabulary set \mathcal{V} ,

$$\Pr(q_1, q_2, \dots, q_k) = \sum_{w \in \mathcal{V}} \Pr(w, q_1, q_2, \dots, q_k) \quad (9)$$

where $\Pr(w, q_1, q_2, \dots, q_k)$ is obtained from Equation 8,

$\Pr(w)$ in Equation 8 can be estimated by summing over all unigram models,

$$\begin{aligned} \Pr(w) &= \sum_{j=1}^p \Pr(M_j, w) \\ &= \sum_{j=1}^p \Pr(M_j) \Pr(w|M_j) \end{aligned} \quad (10)$$

It is not a good idea here to estimate the unigram model $\Pr(w|M_j)$ directly using maximum likelihood estimation, i.e. the number of times that word w occurs in the document j divided by the total number of words in the document, and some degree of smoothing is usually required. One simple smoothing method is to interpolate the probability with a background unigram model,

$$\Pr(w|M_j) = \lambda \frac{c(w, j)}{\sum_{v \in \mathcal{V}(j)} c(v, j)} + (1-\lambda) \frac{c(w, G)}{\sum_{v \in \mathcal{V}(G)} c(v, G)} \quad (11)$$

where G is the collection of all documents, $c(w, j)$ is the number of times that word w occurs in the document j , $\mathcal{V}(j)$ is the vocabulary in the document j , and λ is the smoothing parameter between zero and one.

2.2. Ranking Criterion

While it is tempting to estimate $\Pr(w|R)$ as described in the previous section and re-rank the image list in the decreasing order of $\Pr(D|R)$, there is a potential problem of doing so. Let us look at Equation 2 again. The documents with many words, i.e. long documents, will have

more product terms than short documents, which will result in smaller $\Pr(D|R)$. Therefore, using $\Pr(D|R)$ directly would favor short documents, which is not desirable. Instead, we use Kullback-Leibler (KL) divergence [7] to avoid the short document bias. KL divergence $D(p||q)$ is often used to measure the “distance” between two probability distributions p and q , defined as follows,

$$D(\Pr(w|D_i)||\Pr(w|R)) = \sum_{v \in \mathcal{V}} \Pr(v|D_i) \log \frac{\Pr(v|D_i)}{\Pr(v|R)} \quad (12)$$

where $\Pr(w|D_i)$ is the unigram model from the document associated with rank i image in the list, and $\Pr(w|R)$ is the aforementioned relevance model, and \mathcal{V} is the vocabulary. We estimate the unigram model $\Pr(w|D)$ for each document associated with an image in the image list returned from image search engine, and then calculate the KL divergence between the $\Pr(w|D)$ and $\Pr(w|R)$. If the KL divergence is smaller, the unigram is closer to the relevance model, i.e. the document is likely to be relevant. Therefore, the re-ranking process reorders the list in the increasing order of the KL divergence.

We summarize the proposed re-ranking procedure in Figure 3, where the dashed box represents the “Relevance Model Re-ranking” box in Figure 2. Users input a query consisting of keywords $\{q_1, q_2, \dots, q_k\}$ to describe the pictures they are looking for, and a web image search engine returns a rank list of images. The same query is also fed into a web text search engine, and retrieved documents are used to estimate the relevance model $\Pr(w|R)$ for the query Q . We then calculate the KL divergence between the relevance model and the unigram model $\Pr(w|D)$ of each document D associated with the image I in the image rank list, and re-rank the list according to the divergence.

3. Experiments

We tested the idea of re-ranking on six text queries to a large-scale web image search engine, Google Image Search [10], which has been on-line since July 2001. As of March 2003, there are 425 million images indexed by Google Image Search. With the huge amount of indexed images, there should be large varieties of images, and testing on the search engine of this scale will be more realistic than on an in-house, small-scale web image search system.

Six queries are chosen, as listed in Table 1, which are among image categories in Corel Image Database. Corel Database is often used for evaluating image retrieval [5] and classification [17]. Each text query is typed into Google Image Search, and top 200 entries are saved for evaluation. The default browsing setting for Google Image Search is to return 20 entries per page, and thus 200 entries takes users

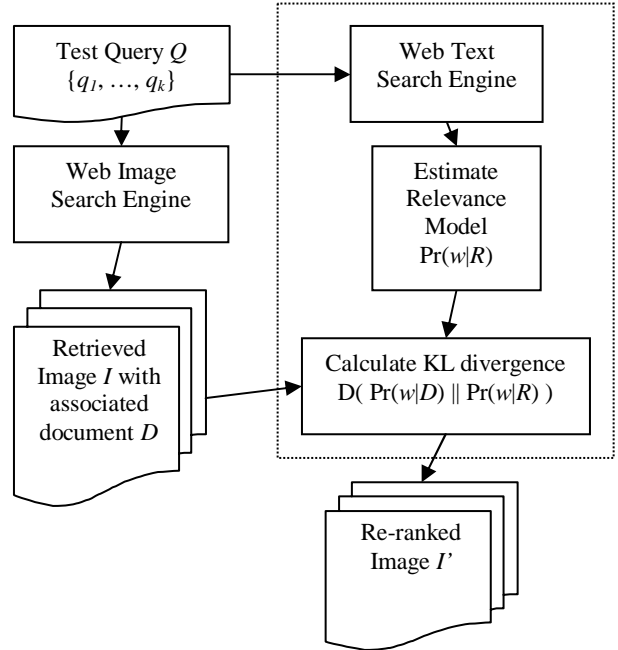


Figure 3. A pictorial summary of relevance model estimation

ten time “Next” button clicks to see all the results, which should reasonably bound the maximum number of entries that most users will check. Each entry in the rank list contains a filename, image size, image resolution, and URL that points to the image. We build a web crawler program to fetch and save both the image and associated HTML document for each entry. After total 1200 images for six queries are fetched, they are manually labeled into three categories: relevant, ambiguous, and irrelevant. An image is labeled as relevant if it is clearly a natural, non-synthesized image with desired objects described by the query, and can be identified instantly by human judges. If the image is obviously a wrong match, it will be labeled irrelevant, otherwise will be labeled as ambiguous. Both irrelevant and ambiguous are considered as “irrelevant” when we evaluate the performance. As shown in the third column of Table 1, the number of the relevant images varies much from query to query, indicating the difficulty of the query.

3.1. Relevance Model Estimation

We also feed the same queries to a web text search engine, Google Web Search [12], to obtain text documents for estimating relevance model. Google Web Search, based on PageRank algorithm[3], is a large-scale and heavily-used web text search engine. As of March 2003, there are more than three billions of web pages indexed by Google Web Search. There are 150 millions queries to Google Web

Table 1. Six search queries

Query No.	Text Query	Number of Relevant Images in Top 200
1	Birds	51
2	Food	117
3	Fish	73
4	Fruits and Vegetables	117
5	Sky	78
6	Flowers	90

Search every day. With the huge amounts of indexed web pages, we expect top-ranked documents will be more representative, and relevance model estimation will be more accurate and reliable.

For each query, we send the same keywords to Google Web Search and obtain a list of relevant documents via Google Web APIs [11]. Top-ranked 200 web documents, i.e. p equals 200 in Equation 8, in the list are further fetched using a web crawler. Before calculating the statistics from these top-ranked HTML documents, we remove all HTML tags, filter out words appearing in the INQUERY [4] stop-word list, and stem words using Porter algorithm [19], which are all common pre-processing in the Information Retrieval systems [2], and usually improve retrieval performance. The relevance model is estimated in the same way described before. The smoothing parameter λ in Equation 11 is empirically set to 0.6.

3.2. Evaluation Metric

Recall and precision are common metrics used to evaluate information retrieval systems. Given a rank list with length n , precision is defined as $\frac{r}{n}$, recall as $\frac{r}{R}$, where r is the number of documents that is truly relevant in the list, and R is the total number of relevant documents in the collection. The goal of any retrieval system is to achieve as higher recall and precision as possible. Here we choose precision at specific document cut-off points (DCP) as the evaluation metric, i.e. calculate the precision after seeing 10, 20, ..., 200 documents.

We choose precision at DCP over traditional Recall-Precision curve is because DCP precision will reflect more closely the browsing behavior of users on the Internet. In the web search setting, users usually have limiting time to browse results, and different methods should be compared after users spend the same efforts of browsing. It should be more reasonable to praise a system that can find more relevant documents in the top 20 results (a specific DCP), rather than at 20% recall which is Precision-Recall curve calculation is based on, because 20% recall can mean different

numbers of documents that have to be evaluated by users. For example, 20% recall means the top 10 documents for the Query 1, but means the top 23 documents for Query 2. In the low DCP, precision is more accurate than recall[13]. Since possible relevant images on the Internet are far larger than we retrieved, 200 documents are regarded as a very low DCP, and therefore only precision is calculated.

3.3. Results

The comparison of performance before and after re-ranking is shown in Figure 4. The average precision at the top 50 documents, i.e. in the first two to three result pages of Google Image Search, has remarkable 30% to 50% increases (recall from original 30-35% to 45% after re-ranking). Even testing on such a high-profile image search engine, the re-ranking process based on relevance model still can improve the performance, suggesting that global information from the document can provide additional cues to judge the relevance of the image.

The improvement at the high ranks is a very desirable property. Internet users are usually with limit time and patience, and high precision at top-ranked documents will save user a lot of efforts and help them find relevant images more easily and quickly.

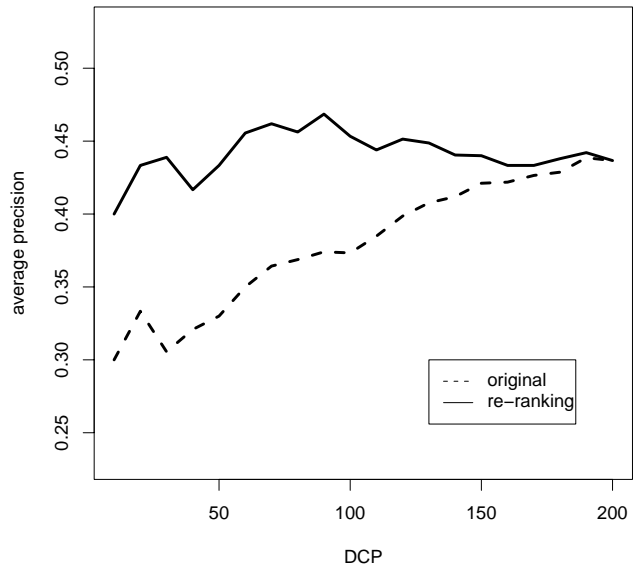


Figure 4. The average precision at DCP over six queries showed re-ranking process made remarkable improvements, especially at the higher ranks

4. Discussions

Let us revisit at the relevance model $\Pr(w|R)$, which may explain why re-ranking based on relevance model works and where the power of the relevance model comes from. In Appendix A, top 100 word stems with highest probability $\Pr(w|R)$ from each query are listed. It appears that many words that are semantics related to the query words are assigned with high probability by the relevance model. For example, in Query 3 “fish”, there are *marine* (*marin* in stemmed form), *aquarium*, *seafood*, *salmon*, *bass*, *trout*, *shark*, etc. In Query 1 “birds”, we can see *birdwatch*, *owl*, *parrot*, *ornithology* (*ornitholog* in stemmed form), *sparrow*, etc. It is the ability to correctly assign probability to semantic related terms that relevance model can make a good guess of the relevance of the web document associated with the image. If the web page contains words that are semantics relevant to the query words, the images within the page will be more likely to be relevant.

Recall we feed the same text query into a web text search engine to obtain top 200 documents when we estimate the co-occurrence probability of the word w and the query Q in Equation 8. These 200 documents are supposed to highly relate to the text query, and words occur in these documents should be very much related to the query. The same idea with a different name called *pseudo relevance feedback* has been proposed and shown performance improvement for text retrieval [22]. Since no humans are involved in the feedback loop, it is a “pseudo” feedback by blindly assuming top 200 documents and relevant. The relevance model estimates the co-occurrence probability from these documents, and then re-ranks the documents associated the images. The relevance model acquires many terms that are semantics related the query words, which in fact equals to *query expansion*, a technique widely used in Information Retrieval community. By adding more related terms in the query, the system is expected to retrieve more relevant documents, which is similar to use relevance model to re-rank the documents. For example, it may be hard to judge the relevance of the document using single query word “fish”, but it will become easier if we take terms such as “marine”, “aquarium”, “seafood”, “salmon” into consideration, and implicitly images in the page with many fish-related terms should be more likely to be real fish. The best thing about relevance model is that it is learned automatically from documents on the Internet, and we do not need to prepare any training documents.

5. Conclusions and Future Works

Re-ranking web image retrieval can improve the performance of web image retrieval, which is supported by the experiment results. The re-ranking process based on rele-

vance model utilizes global information from the image’s HTML document to evaluate the relevance of the image. The relevance model can be learned automatically from a web text search engine without preparing any training data.

The reasonable next step is to evaluate the idea of re-ranking on more and different types of queries. At the same time, it will be infeasible to manually label thousands of images retrieved from a web image search engine. An alternative is task-oriented evaluation, like image similarity search. Given a query from Corel Image Database, can we re-rank images returned from a web image search engine and use top-rank images to find similar images in the database? We then can evaluate the performance of the re-ranking process on similarity search task as a proxy to true objective function.

Although we apply the idea of re-ranking on web image retrieval in this paper, there are no constraints that re-ranking process cannot be applied to other web media search. Re-ranking process will be applicable if the media files are associated with web pages, such as video, music files, MIDI files, speech wave files, etc. Re-ranking process may provide additional information to judge the relevance of the media file.

References

- [1] Altavista image. <http://www.altavista.com/image/>.
- [2] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, 1999.
- [3] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of 7th International World-Wide Web Conference*, 1998.
- [4] J. P. Callan, W. B. Croft, and S. M. Harding. The IN-QUERY retrieval system. In *Proceedings of 3rd International Conference on Database and Expert Systems Applications*, 1992.
- [5] C. Carlson and V. E. Ogle. Storage and retrieval of feature data for a very large online image collection. *Bulletin of the Technical Committee on Data Engineering*, 19(4), 1996.
- [6] Corbis. <http://www.corbis.com/>.
- [7] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.
- [8] Ditto. <http://www.ditto.com/>.
- [9] Getty images. <http://creative.gettyimages.com/>.
- [10] Google Image Search. <http://images.google.com/>.
- [11] Google Web APIs. <http://www.google.com/apis/>.
- [12] Google Web Search. <http://www.google.com/>.
- [13] D. Hull. Using statistical testing in the evaluation of retrieval experiments. In *Proceedings of the 16th SIGIR Conference*, pages 329–338, 1993.
- [14] J. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of ACM*, 46(5):604–632, 1999.
- [15] V. Lavrenko and W. B. Croft. Relevance-based language models. In *Proceedings of the International ACM SIGIR Conference*, 2001.

- [16] R. Lempel and A. Soffer. PicASHOW: Pictorial authority search by hyperlinks on the web. In *Proceedings of the 10th International World-Wide Web Conference*, 2001.
- [17] O. Maron and A. L. Ratan. Multiple-instance learning for natural scene classification. In *Proceedings of the 15th International Conference on Machine Learning*, pages 341–349, 1998.
- [18] PicSearch. <http://www.picsearch.com/>.
- [19] M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.
- [20] J. R. Smith and S.-F. Chang. VisualSEEK: a fully automated content-based image query system. In *Proceedings of ACM Multimedia*, 1996.
- [21] C. J. van Rijsbergen. A theoretical basis for the use of co-occurrence data in information retrieval. *Journal of Documentation*, 33:106–119, 1977.
- [22] J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th Annual International ACM SIGIR Conference*, 1996.

A. Appendix

Each column contains 100 word stems with the highest probabilities assigned by the relevance model $\Pr(w|R)$. Words that occur in more than three queries are considered as web-specific stopwords (e.g. “web”, “website”) and thus are not listed.

Birds	Food	Fish	Fruits and Vegetables	Sky	Flowers
club	safeti	market	veget	telescop	gift field
carolina	recip	wildlif	improv	vanilla	order
speci	2000	sydney	garden	digit sea	plant
photo	polic	speci	passion	weather	garden
pet tour	law	big	plant	red	deliveri
publish	futur	famili	grow	interact	florist
canada	summit	reel	tree	citi voic	send
wildlif	nutrit	marin	loom	blue	2003
studi	intern	hunt	crop	calendar	bouquet
market-	cook	english	organ	casino	wed
plac	institut	learn	seed soil	snowwar	show
aviat	agricul-	law	fresh	as-	rose
ameri-	tur fda	fisheri	grower	tronomi	essenc
can	eat	name	individu	big star	shop
pixar	organ	boat	recip	limit	fuzz
north	health	depart	ontario	check	acid uk
song	term	aquar-	agricul-	night	basket
bird-	ifpri	ium idea	tur	earth	offici
watch	report	com-	associ	express	deliv
start	fat	miss	tomato	sport	design
wild	industri	question	varieti	skan	arrang
jpeg	articl	conserv	appl	infin	special
warbler	public	report	veggi	tour	film
famili	wfp	angler	food	plug	hospit
owl	scienc	introd-	tropic	imag	floral
parrot	tip	uct net	special	look	case
learn	nation	leader	gift pest	space	press
natur	condit	philosophi	com-	databas	virtual
anim	consum	neav	merci	light	love 95
support	educ	chart-	sweet	observ	mound
area	html	hous	contain	version	fresh
info	market	public	long eat	00	town dri
magazin	11 label	lake	small	moon	custom
or-	pyramid	manag	mission	system	cham-
nitholog	group	educ fly	water	comput	pagn
town	novemb	nation	part	deep	super
life	usda pdf	top	topic	win stig	drink
foundat	co serv	articl	florida	ami	east pc
window	2001	water	good	question	secur
america	develop	seafood	fertil	woman	hothous
white	fruit	salmon	season	phe-	birthdai
com-	daili	bass	industri	nomen	select
mon	read	trout	requir	bigsky-	occas
version	meat	river	harvest	band	thank
sparrow	compani	com-	potato	2001	price
british	com-	ment	pear cal-	object	societi
black	mun	photo	ifornia	planet	2000 99
long	australia	polic	fact	send	lei ship
south	univers	shark	spring	survei	call
hous bill	full	aquacul-	develop	show	qualiti
red nest	import	tur	leav	footbal	anim
trip	chef 02	freshwat	green	as-	look
celebr	veget	index	produc	tronom	beauti
	restaur	check	select	studio	chicago
	fact	design	univers	reach	info
		plan	bean	org	offer
		group	flower	down-	power
			area	load dec	remedi
			diseas	10 locat	postcard
			label	feed-	fairi
			basket	back	tropic
				pearl	
				term	