

# Method of acquiring web features and its application in web search

Xue Yewei Shen Junyi Zhang Yun Bao Junpeng

(Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an 710049, China)

**Abstract:** Focusing on the problem that it is hard to utilize the web multi-fields information with various forms in large scale web search, a novel approach, which can automatically acquire features from web pages based on a set of well defined rules, is proposed. The features describe the contents of web pages from different aspects and they can be used to improve the ranking performance for web search. The acquired feature has the advantages of unified form and less noise, and can easily be used in web page relevance ranking. A special specs for judging the relevance between user queries and acquired features is also proposed. Experimental results show that the features acquired by the proposed approach and the feature relevance specs can significantly improve the relevance ranking performance for web search.

**Key words:** web search; relevance ranking; retrieval effectiveness

This paper focuses on large scale web page relevance ranking problems, including the relevance level judgment and relevance feature acquisition. Relevance ranking is a central problem for web search, because the efficiency of a web search engine is mainly evaluated by the relevancy of its search results. The web search means finding all the web pages relevant with the queries given by users, ranking them according to some relevance measures, and returning the results in an orderly fashion. Obviously, how to order the results is a key issue for the search system, because it can directly influence user impressions.

Most of the previous relevance ranking approaches directly used traditional information retrieval methods designed for text documents<sup>[1]</sup>, which do not sufficiently consider the specialty of the web pages and are directly used on web texts. Along with the development of search engines, more and more researchers have noticed that besides the body texts of web pages, there are other important resources to express the relevance. In 1998, Brin and Page first used the anchor text in their search engine, which is now called Google, and obtained great success<sup>[2]</sup>. Then, more and more researchers paid attention to mine new relevance materials for search engines. Web search can be viewed as a special case of document retrieval. A web page normally can be seen consisting of four fields: body, title, anchor, and URL. Although there are other useful fields, for example, snippet, click-through data, abstract, etc., they are not often used. Previously, researchers tried to measure the relevance between the query and each of the fields and then use them to

assist the relevance judgment process for the whole web page<sup>[3-5]</sup>.

In information retrieval (IR), many models have been proposed to measure the relevance of a document with respect to a query being searched. One approach is to take into consideration the frequencies of query words in the document, which is often called the bag-of-words model (because they often ignore the dependencies among the query words). The idea is that the more frequently the query words occur in the document, the more likely the document is relevant. The most famous model is BM25<sup>[6]</sup>, which is widely used in various information retrieval tasks.

We notice that the title, anchor, and URL of a web page usually represent a name of the page, and they should be more useful for representing relevance between the query and the web page. The body text, which is often used in many web search algorithms, has been verified to be effective for information retrieval and web search. Queries, users submitted to search engine to express their information needs, can also be seen as a kind of name of the web pages they want to obtain. When the search engine ranks the target web pages, it actually finds relevance features on each web page and ranks the web page according to the relevance level between its features and the given query.

The problem is finding a novel approach to acquire features from the fields listed above and utilize these features directly to rank the objective web pages. To address the problem, we propose a novel rules-based approach to acquire the relevance features and a special spec to directly judge the relevance between the query and acquired features. After we obtain the relevance measure in the separated features, we use a simple linear fusion function on them to obtain a final relevance measure for the whole web page.

## 1 Acquiring Web Features

### 1.1 Problems and definitions

Most of the current web search ranking models directly using the resources come from different web fields. But these fields have very large differences among them. For example, the body field is much longer than any of the other fields and the URL field is completely different from the title and anchors. So we think that once we can get some unified relevance resources from these different fields, which we call web features in this paper, then we can further simplify the ranking process and help to improve web search performance. The problem is how to define the relevance feature in the web pages. We think that the relevance feature is the most conspicuous description of the content of a corresponding field and it must meet the following conditions:

1) Content: The extracted feature must have the same meaning as that of the original field. That is to say, the process, acquiring the feature from a corresponding field,

Received 2008-04-15.

**Biographies:** Xue Yewei (1980—), male, graduate; Shen Junyi (corresponding author), male, professor, jyshen@mail.xjtu.edu.cn.

**Foundation item:** The National Natural Science Foundation of China (No. 60673087).

**Citation:** Xue Yewei, Shen Junyi, Zhang Yun, et al. Method of acquiring web features and its application in web search[J]. Journal of Southeast University (English Edition), 2008, 24(3): 330 – 334.

cannot make any changes in the key words which may hold the main concept of the field.

2) Size: The extracted feature may be a composite of many noun phrases and have a small length compared with the original field. The language sentence is not a good choice, because most of the queries that the user submitted to the search engine are very short noun phrases and not sentences<sup>[7]</sup>.

3) Symbol: Many fields contain some symbols, parts of them are only used for sentence segments. But in web pages, some symbols have special meanings at some times; for example, the “&” in AT&T cannot be treated as a segment symbol and the “+” in C++ must also be retained in the extracted relevance feature.

4) Exception: Many web page fields contain certain content to represent some special states, for example, the “Untitled” appearing in the web title field means that the web page has no title in the title field. These exceptions must be taken into account when we process the field.

In short, the relevance feature means that there is some noun phrase extracted from the various web fields which can stand for the content of the original source. The relevance features can exist in any web page resources, just like body, title, anchor and so on. In this paper, for simplification, we only consider acquiring relevance features from body, title, anchor text, and URL, because they are all very frequently used in current web search tasks.

1.2 Architecture of the AcWF system

We design a rule-based system to acquire the relevance features from web fields automatically, referred to as AcWF. When a web page is stored and indexed by a search engine, various resources concerning this page are collected, including the web content, the link relationship with other pages, web address, etc. After this step, the AcWF is used on these resources to acquire the relevance features. The extracted relevance features can express the content of the original resources, so they can be used directly in web search ranking models instead of the complicated original resources.

Fig. 1 shows the structure of the AcWF. The AcWF works as follows:

- 1) The AcWF system gets the initial data from a search engine, which includes the web page content and the link relationship description. The URL field can be obtained directly from the search engine data. The web page content data are sent to the web parser and processed into DOM tree nodes by the HTMLParser, from which we can obtain the title field and the DOM tree data for the body field.
- 2) The anchor field data can be obtained by the link ana-

lyzer, which is used to analyze the link relationship description data and extract anchor texts from web pages which point to the current web page. It also identifies different anchor texts and calculates the appearance frequency for each anchor text. Then the “anchor” field is composed of all the “anchor” texts and their frequencies.

3) After the title, anchor, body, and URL fields data are obtained by the above steps, they will be sent to the feature extraction processor. This unit processes each field according to well-defined rules and stores in a rules set and then generates the relevance features for corresponding fields. But for some fields, for example, anchor and body, the processor may generate more than one feature, so we can get more than four features finally. Of course, if the field data are missing, the AcWF will only output a null string accordingly.

1.3 Rules used in the AcWF

The rules used in the AcWF system can be classified into two types: special rules and common rules. The former are used only on certain fields and the latter are suitable for every field. The three fields that need to use special rules are URL, body, and anchor. A simple description of the special rules can be listed as follows:

1) URL: The URL is the Internet address for a web page. Most of the URL contains some useful information for web content, for example, the website name, the web content name, etc. But the useful information is not distributed evenly in the whole URL string. Usually, the website information is often located in the beginning part and the content description is located in the end part; the middle part shows the access path. So, most of the time, we need to adjust the order of these substrings according to their importance and delete the useless substrings before using the common rules on the URL field, for example, the “http://” and the “.htm” can be deleted.

2) Body: For the body field, we obtain the DOM tree structure by the HTMLParser. Based on the DOM tree, we can conduct two further processes: one is to divide the body text by sentences and the other is to extract content title or highlight texts in the body. For simplicity, we only choose the sentences which have different formats from the texts surrounding them or with the largest font size as the relevance feature candidates for the body field. More accurate approaches can be found in Ref. [8].

3) Anchor: From the anchor field data, we know all the anchor texts and their frequencies for a certain web page. But whether all the anchor texts can be treated as relevance features, of course, the answer is not. According to our experience, we choose five anchor texts with the most frequency. If there are fewer than five anchor texts, we select all of them. Please notice that, after we select the anchor text candidates, we do not need to care about their frequencies, that is to say, we think they have the same importance (Actually, although the anchor texts for one web page have different contents, they are all descriptions for the same web page, only given by different persons, so if they have high frequencies, they are all true.).

After using the special rules, we obtain the feature candidates for the fields, and the AcWF system will further

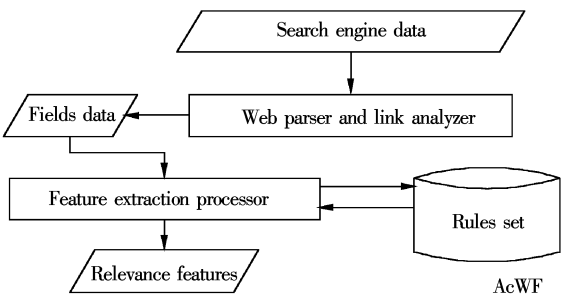


Fig. 1 Architecture of the AcWF system

process them according to the following common rules:

- 1) Deletion rules: Delete the following strings contained by the candidates: “home”, “homepage”, “homepage of”, “welcome”, “welcome to”, etc. , because these strings have no identification ability and cannot provide any useful information for ranking. For the same reason, delete the following strings if they are located after“.” symbol: “com”, “edu”, “gov”, etc. and “cn”, “jp”, “uk”, etc.
- 2) Symbol rules: Finding two special symbols: “:” and “-”. If they are located in the middle of the candidates, retain the head part of the candidates and delete the others. In such a case, the head part often contains the most important information and the following string often expands the content. Another process rule is that if it is not in conflict with the exception rules below, it then replaces all the symbols with a space.
- 3) Exception rules: For “+” and “#”, check the characters preceding them, if they are “c++”, “j++”, etc. special phrases, retain them; otherwise, delete them. For & , \_ , @ and % , just retain them. Retain the “.” between two numbers.

By the above processes, the AcWF can output all the relevance features for web pages, these features have the same content meaning as the original fields, but they are shorter and better for matted. So it can be directly used to measure the relevance between the web pages and queries.

2 Feature Relevance Level

Although many researchers have tried to utilize multi-field information to improve web search performance, there is still not a relevance measure spec for directly judging the relevance level between a query and web fields. All the current models only use the relevance level spec defined to measure the whole web page content, which most of the time actually stands for the body’s content. Obviously, it may cause mistakes when such a spec is used for measuring other fields; for example, someone may judge a web page relevant to a query, and he may mean that the main content fits the query. But another one may only have interest in some part of the content of the page, and this description often exists in the anchor texts pointing to the page. In this case, if we

judge the relevance between the query and the anchor texts, it is difficult to understand how to use the spec for a web page. So finding a spec for judging the relevance of fields, features, or any other short strings is a critical task.

From another aspect, given a query and a short string, whether it is a sentence or not, one can easily judge the relevance between them and give a hierarchical result. Although it is also a subjective process, the relevance judgment for a short string is much easier than that for a whole web page, because the former only needs to consider the content of both the query and the target string.

The spec currently used for web page level relevance judgment is of different types: one type is a two-level spec, which only contains two different levels relevant and irrelevant. The two-level spec is often used in traditional information retrieval tasks and previous web search tasks. Another type is a multi-level spec, which contains more than two relevance levels. The TREC<sup>[9]</sup> provided their . gov2 data with three relevance level judgments ( highly relevant, relevant and irrelevant) and some commercial search engines often use a spec with seven or more levels. All these events prove that it is not very easy for judging the relevance between a query and a whole web page. The situation for relevance features or other short strings is quite different.

We conduct user studies to judge which number of levels is fit for measuring the relevance between the feature and query. We select some real queries and corresponding returned web pages supported by a very famous commercial search engine and ask seven graduate students to judge the relevances between the selected queries and the relevance features extracted from the web pages. All of the assessors are graduate students of computer science in several very famous universities and skilled with the Internet, search engine, and languages. The judging process is independently conducted among the assessors and finally we find that the five-level judgment obtains the most agreement among the assessors. After carefully studying the judging results, we propose a spec with five relevance levels to directly measure the relevance among the queries and features. Tab. 1 shows the details of the five-level relevance spec.

Tab.1 Five-level spec for feature relevance measure

Level	Name	Description
0	Exact match	The feature string is exactly the same as the query string. They represent the same concept.
1	Approximately exact match	The feature string has the same or a similar meaning as the query string. They match each other approximately.
2	Relevant partial match	The feature string partially matches the query string. It contains a concept that is highly relevant to the query, for example, a sub-concept or a super-concept.
3	Irrelevant partial match	The feature string partially matches the query string. However, the match is superficial. There is no or a weak relevance between the two.
4	Non match	The feature string does not match the query string. They are not conceptually related.

By the spec, we can directly tune the parameters to the feature or the single field when we use some machine learning models that have the parameters necessary to fit the real data. But the traditional spec for the whole page may cause a misfit for parameters tuning on to single field data.

3 Application in Web Search

First, we recall the search engine’s work process. When a

user submits a query to the search engine, it records the query content and finds the object web pages in its data store. After finding a set of candidates, the search engine needs to rank them according to the relevance among the candidates and the submitted query. So we can find that the core problem is how to measure the relevance between the query and the web pages selected as candidates. Most of the ranking models utilize the term frequency information, which is

based on a simple assumption that the more the query terms appear in the document, the more relevance they possess. The traditional ranking models are often used in the body text of the web pages and make some sense. But the results are not very good when they are used in other fields. That is just because the traditional relevance judgment is generated by the content of the body text. It fits the distribution on the body field but does not exactly fit other fields. To solve this problem, we think it is better to tune the model separately on each feature or field according to the feature level relevance spec and then combine the separate scores into a final ranking result for the whole web page.

Considering a web page document  $d$  contains several fields  $f$ , when we measure the relevance between query  $q$  and web page  $d$ , we always take into account the whole page's information:

$$d: = f_1 + f_2 + \dots + f_k \quad (1)$$

This means that the concept of the whole web page needs to consider all the field contents.

In the traditional approach, when the relevance ranking model  $g$  is used, it will be tuned to fit the whole web page document and the final ranking function can be shown below:

$$R(d, q) = wg(d, q) = \sum_i g_d(f_i, q) w_i \quad (2)$$

The  $g_d$  in the above function denotes the ranking model  $g$  tuned by the web level relevance spec.

In our approach, the ranking model is tuned to each field according to the feature level relevance spec. Then the ranking function can be presented as follows:

$$R(d, q) = \sum_i g(f_i, q) w_i = \sum_i g_{f_i}(f_i, q) w_i \quad (3)$$

Because our contributions are the extracted features and the feature level relevance judgments, we simply use the linear combination for generating the final results, which is also widely used in most of the IR tasks. Also researchers have proposed some non-linear combination methods<sup>[10]</sup>, but it is not our issue, so we do not mention it in this paper.

## 4 Experiments

### 4.1 Datasets and measures

We hypothesize that the use of the relevance feature and feature level relevance spec can further improve the ranking performance for web search. Since the main advantage of the proposed system is to take into account relevance measures for the separate fields and features, we expect the new approach to perform best on collections with multi-fields. Indeed, the algorithm was originally designed for corporate collections containing various web page fields, such as title, anchor, body, URL and so on.

We use the 2002 TREC web-track crawl of the .gov domain (which we will refer to as .Gov) and the 2004 TREC Terabyte Track<sup>[9]</sup> (which we will refer to as .Gov2) for evaluation. These two collections expose all the types of fields we mentioned above. Some details can be found in Tab. 2. The queries we use with the .Gov dataset are mixed

by TREC—2004 NP and HP tasks, and for .Gov2, we directly use the 701 to 750 topics listed in TREC 2004.

**Tab. 2** Test collection characteristics

Corpus	Number of pages	Size/Gbit	Relevance level
. Gov	$> 1.05 \times 10^6$	18	2
. Gov2	$> 25.2 \times 10^6$	427	3

Due to space limitations, we only choose mean average precision (MAP)<sup>[11]</sup> and precision at rank = 10 (Prec@10)<sup>[11]</sup> as primary measures for representing our experimental results. Similar experimental results have been observed using other measures (e. g. MRR, P@5, etc. ).

### 4.2 Evaluation methods

We choose the Okapi BM25 model<sup>[6]</sup> as the basic relevance ranking function to evaluate the proposed approach and compute the relevance between the query and a single field or feature.

$$S = \sum_{i \in q} \frac{(k_1 + 1)tf_i}{k_1 \left( (1 - b) + b \frac{dl}{avdl} \right) + tf_i} \log \frac{N - df_i + 0.5}{df_i + 0.5} \quad (4)$$

where  $i$  denotes a word in the query  $q$ ;  $tf_i$  and  $df_i$  are term frequency and document frequency of  $i$ , respectively;  $dl$  is document length, and  $avdl$  is the average document length;  $k_1$  and  $b$  are parameters.

For a score in a single field or feature, we first optimize  $k_1$  and  $b$  on each collection and each field separately, considering as a collection only the fields being scored. The only difference between our approach and the baseline method is that we optimize parameters directly using the feature level relevance judgment, but the baseline uses the web page level judgment. This requires a single optimization of two parameters in each field, and we use the grid search method to find the optimized parameters.

For final scores combined with various fields, field weights  $w_i$  are optimized in the same way for both the baseline method and our approach. We set the body weight to 1 and optimize the remaining weights. The optimization cost is the same for both methods; a single optimization of three parameters. The choice of the initial field weight does not affect the final results.

In the experiments, we conduct 5-fold cross validation on the query set, and thus all the results are averaged over five trials. The parameters are tuned to the training set and then the optimized parameters are applied to the testing set.

### 4.3 Experimental results

In the following text, we refer to the results of the baseline combined on the four fields as BASE and the results given by our approach as AWF. The results are shown in Fig. 2 and Fig. 3.

From the results we can see that the combination of the multi-fields can indeed improve the ranking performance. However, AWF can further improve performance significantly ( $p$ -value  $< 0.05$ , MAP) on both datasets, which strongly indicates that our approach is very useful for ranking in web

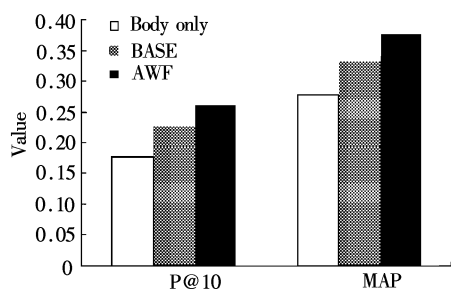


Fig. 2 Ranking performance on Gov data

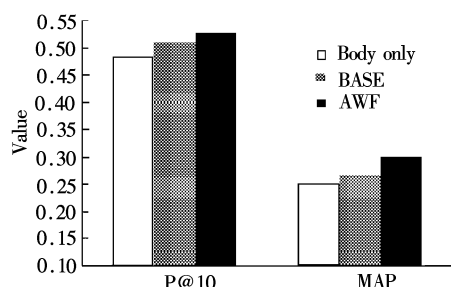


Fig. 3 Ranking performance on Gov2 data

search. Due to space limitations, we only show the results for two measures, but similar experimental results have been observed using other measures (e. g. MRR, P@ 5, etc. ).

## 5 Conclusion

In this paper, we propose a novel approach to address the problem of multi-fields relevance ranking for web search. The proposed approach in this paper includes a rule-based system to automatically acquire the relevance features from web data and a special spec used for judging the relevance between the extracted feature and the query. We conduct the experiments with two famous public TREC datasets to evaluate the performance of our approach. Experimental results show that our approach can significantly outperform the baselines, indicating that it is very useful for relevance ranking. Future work will be conducted on the following subjects:

1) How to extract the relevance feature by a machine learning method; 2) Non-linear combination approach for multi-fields relevance ranking; 3) Testing the performance based on real search engine data.

## References

- [1] Baeza-Yates R A, Ribeiro-Neto B A. *Modern information retrieval* [M]. New York: Addison-Wesley, 1999: 27-86.
- [2] Brin S, Page L. The anatomy of a large-scale hypertextual web search engine [J]. *Computer Networks and ISDN Systems*, 1998, **30**(1/2/3/4/5/6/7): 107-117.
- [3] Ogilvie P, Callan J. Combining structural information and the use of priors in mixed named-page and homepage finding [C]//*Proc of TREC'03*. Gaithersburg: NIST Special Publication 500-251, 2003: 177-184.
- [4] Song R, Xin G, Shi S, et al. Exploring url hit priors for web search [C]//*Proc of ECIR'06*. Berlin: Springer, 2006: 277-288.
- [5] Westerveld T, Kraaij W, Hiemstra D. Retrieving web pages using content, links, urls and anchors [C]//*Proc of TREC'01*. Gaithersburg: NIST Special Publication 500-249, 2001: 663-672.
- [6] Robertson S E, Walker S, Hancock-Beaulieu M. Experimentation as a way of life: Okapi at TREC [J]. *Information Processing and Management*, 2000, **36**(1): 95-108.
- [7] Mittal V, Baluja S, Sahami M. Google tutorial on web information retrieval [C]//*Proc of RIAO 2004 Coupling Approaches, Coupling Media and Coupling Languages for Information Retrieval*. Avignon, France, 2004.
- [8] Xue Y, Hu Y, Xin G, et al. Web page title extraction and its application [J]. *Information Processing and Management*, 2007, **43**(5): 1332-1347.
- [9] TREC. TREC data information [EB/OL]. (2004-07-15) [2008-02-20]. <http://trec.nist.gov/data.html>.
- [10] Shi S, Song R, Wen J. Latent additivity: combining homogeneous evidence, MSR-TR-2006-110 [R]. Microsoft Research, 2006.
- [11] Craswell N, Hawking D. Overview of the TREC-2002 web track [C]//*Proc of TREC'02*. Gaithersburg: NIST Special Publication 500-251, 2003: 78-92.

# 网页特征获取方法及其在网页搜索中的应用

薛晔伟 沈钧毅 张云 鲍军鹏

(西安交通大学计算机科学与技术系, 西安 710049)

**摘要:**针对大规模网页相关性排序工作中使用的多来源网页信息形式多样、利用困难的问题,提出了一种新的自动网页特征获取方法.该方法利用一组事先定义好的规则自动地从网页中获取相关性特征,这些特征可以有效地表达网页的实际内容并改善搜索引擎的排序性能.该方法所获取的网页相关性特征具有格式统一、噪声数据少的特点,能够非常方便地应用于网页的相关性排序.为了评价网页特征和用户查询之间的相关性,还提出了一个特征级别的相关性判定标准.最后,实验结果证明了所提出的特征获取方法和特征相关性等级判定标准对于提升搜索引擎的排序性能具有显著的作用.

**关键词:**网页搜索;相关性排序;检索效率

**中图分类号:**TP391