Article ID:

Automatic User Goals Identification Based on Anchor Text and Click-through Data

YUAN Xiaojie, DOU Zhicheng, ZHANG Lu, LIU Fang
College of Information Technical and Science, Nankai University, Tianjin 300071, China

Abstract: Understanding the underlying goal behind a user's Web query has been proved to be helpful to improve the quality of search. This paper focuses on the problem of automatic identification of query types according to the goals. Four novel *entropy-based* features extracted from anchor data and click-through data are proposed, and a SVM classifier is used to identify user goal based on these features. Experimental results show that proposed entropy-based features are more effective than those reported in previous work. By combining multiple features the goals for more than 97% queries studied can be correctly identified. Besides these, this paper gets following important conclusions: First, anchor-based features are more effective than click-through-based features; Second, number of sites is more reliable than number of links; Third, click-distribution-based features are more effective than session-based ones.

Keywords: query classification; user goals; anchor text; click-through data; information retrieval
CLC Number: TP391

0 Introduction

There have been growing interests in understanding the underlying goals behind a user search. Border[1] , Rose and Levinson[2] , and Lee et al.[3] have found that the goal of a user underlying a query can be classified into at least two categories: *navigational* and *informational*. When issuing a navigational query, the user wants to reach a particular website or page he already has in mind. For example, navigational query "sina" is usually used to redirect to website www.sina.com.cn. While for an informational query, the user does not have a particular page in mind or trends to visit multiple pages to learn about a topic. "EM algorithm" is an informational query. The goal of this query is to learn about EM algorithm and the user does not have a preferred Web page in mind. He may need to explore several relevant web pages until he gets what he wants.

It is obvious that informational and navigational queries have different characters in ranking. We could adapt different retrieval algorithms according to the type of a query and improve overall search quality. Craswell[4] and Westerveld et al.[5] demonstrated that it is feasible to improve search performance by applying specialized ranking strategies for informational and navigational queries. Kang and Kim[6] revealed that for informational queries, we could emphasize content information. While for navigational queries, we should emphasize anchor, link, and URL information. Therefore, it is worthwhile to automatically identify the type of a query first, and this is the main task of this paper.

There has been some previous work on automatic identification of user goals. Kang and Kim[6] proposed to use the occurrence patterns of query terms in root documents, titles, and anchor text to identify query types. They proposed three features including anchor usage rate, query term distribution, and term dependence. Lee et al.[3] pointed out that the three features proposed by Kang and Kim[6] were not very effective in predicting user goals. They proposed to use user-click behavior and anchor-link distribution. They experimented with the median of click distribution and anchor-link distribution. Liu et al.[7] proposed two features nCS and nRS extracted from query sessions in click-through data to accomplish this task. Experimental results showed that by combining these two features with click-distribution features introduced in [3] , better performance could be achieved.

In this paper, we also use anchor data and click-through data to identify query goals. We propose four novel *entropy-based* features: click entropy, domain click entropy, link entropy, and site entropy. We use the support vector machines (SVM) algorithm [8] to identify user goals based on these features. We also implement all the features introduced by Lee et al.[3] and Liu et al.[7] because to our best knowledge they are currently the most effective features. Experimental results show that our proposed features are more effective than prior ones. We also reveal that when using anchor information, the anchor-site distribution is more effective and stable than the anchor-link distribution. We also show that by combining multiple features, we can correctly identify the goals for more than 97% of the queries studied.

Please note that the taxonomies proposed by Border[1] , and Rose and Levinson[2] have a third category besides navigational queries and informational queries—transactional queries[1] or resource queries[2] . Lee et al.[3] and Liu et al.[7] merged transactional queries and informational queries, and only considered

Received Date: 2008-5-1
Foundation Item: Supported by the Tianjin Applied Fundamental Research Plan (No. 07JCYBJC14500)
Biography: YUAN Xiaojie (1963-), female, Professor, Ph. D., research direction: database, XML, and information retrieval. Email: yuanxj@nankai.edu.cn. Tel: 022-23508468

navigational and informational (and/or transactional) categories. In this paper, we also use this taxonomy because informational and transactional queries have similar characters.

1 Query Type Identification Features

In this section, we propose four novel features extracted from anchor text and click-through data: click entropy and domain click entropy extracted from click-through log, and link entropy and site entropy extracted from Web page anchor data.

1.1 Click Entropy and Domain Click Entropy

User's click behavior on the returned results is very useful evidence for identifying the user's goal behind a query. If in the past users have consistently clicked on one or a few Web pages for a query, this query is very likely to be a navigational query. On the contrary, if users have clicked on many results, it might be an informational query.

For a given query *q*, suppose C_q is the collection of clicked Web pages for this query. For each Web page *p* in C_q, the corresponding click probability $P_{\text{click}}(p|q)$ is defined as:

$$P_{\text{click}}(p\,|\,q) = \frac{\#\text{clk}(q,p)}{\#\text{clk}(q)} = \frac{\#\text{clk}(q,p)}{\sum_{i\in C_q}\#\text{clk}(q,i)}$$

Here #clk(*q, p*) is the number of clicks on Web page *p* for query *q*, and #clk(*q*) is the total number of clicks for this query. We define the click entropy (**ClickEntropy**) of query *q* as the following.

$$\text{ClickEntropy}(q) = -\sum_{p\in C_q} P_{\text{click}}(p\,|\,q)\log P_{\text{click}}(p\,|\,q)$$

If all users click only one identical page on query *q*, click entropy of query *q* will be 0. Smaller click entropy means that the majority of users agree with each other on a small number of web pages. Large click entropy indicates that many web pages are clicked for the query.

By observing a real-world click-through log (We will introduced the data in Section 3), we find click-through data contain much noise. Some users are very lazy. They usually use a short domain name to reach one of its sub domains (because some popular sub domains are usually also ranked to the first page). For example, in the log data, we find only about 40% of clicks for the query "17173" are made on the URL "17173.com". Almost all left 60% clicks are made on its 40 sub-domain URLs, among which "download.17173.com/" receives about 20% clicks. The clicks on these sub domains will increase the click entropy value and reduce the identification accuracy. In order to solve this problem, we merge all URLs within a same domain and sum up all clicks on these URLs as the total clicks on the domain. We then re-calculate click entropy based on the clicks on domains, and name this feature as **DomainClickEntropy**.

1.2 Link Entropy and Site Entropy

Another feature that we may use is the destinations of the links with the same anchor text as the query. Anchor text can be thought as query candidate given by webmasters. If most webmasters use an anchor text to link to a same website, this anchor text is very likely to be a navigational query which will be issued by users.

For a given query *q*, we find all the anchors appearing on the Web that have the same text as the query, and extract their destination URLs. We denote the collection of destination Web pages for this query with L_q, and use #links*(q, p)* to denote the number of links which link to the Web page *p* with anchor text *q*. For each Web page *p* in L_q, we define the percentage of links for page *p* as:

$$P_{\text{link}}(p\,|\,q) = \frac{\#\text{links}(q,p)}{\#\text{links}(q)} = \frac{\#\text{links}(q,p)}{\sum_{i\in L_q}\#\text{links}(q,i)}$$

We define the link entropy of query *q* as the following.

$$\text{LinkEntropy}(q) = -\sum_{p\in L_q} P_{\text{link}}(p\,|\,q)\log P_{\text{link}}(p\,|\,q)$$

If all links with anchor text *q* point to one identical page, obviously link entropy of *q* is 0. A smaller link entropy value for a query (anchor text) means that the majority of pages use the anchor text to link to a same Web page. When a user issues this anchor text as query, he may also want to reach this authoritative Web page. Obviously such a query is a navigational query. A larger link entropy value indicates that many web pages are linked with the anchor text and the anchor text is more likely to be about a topic but not a website. We denote this feature with **LinkEntropy** in the remaining parts of this paper.

In the LinkEntropy feature, we use the number of links which contain the anchor text. In fact, this may have some issues. A webmaster could generate a large number of pages within a website and link them to any page with the same anchor text. This could spam the link distribution of this anchor text. Please see the example given in Table 1. "起点" is the Chinese name of a famous online writing and reading website (www.cmfu.com). Unfortunately, there are many other Web pages which also have a large number of links with this anchor text in the Web. After checking the content of these pages, we find most of them are index pages for a php programming language manual[1]. The manual contains many Web pages and each page has a link to the index page with this anchor text (because it means "start point" in Chinese). If we calculate entropy based on link distribution for this anchor text, we will get a large value because there are many Web pages which have a large link percentage. Obviously this is problematic.

To solve this problem, we propose to use the number of sites which point to the destination URL instead of the number of links. Compared with the number of links, the number of sites is more reliable and stable. A popular website usually has a large portion of sites linking to it. And it is usually more difficult and costly to add spamming links in a large number of websites. Table 1 shows that there are more than 95% sites linked to website cmfu.com, while all

[1] please see http://www.netbei.com/online/phpmanual/index.html for example

Table 1 Top 5 pages linked by anchor text “起点” sorted by number of links

Rank	URL	#Links
1	http://php.phpcms.cn/book/php_manual/index.html	3,336
2	http://www.cmfu.com/index.asp	2,143
3	http://www.cmfu.com/	1,268
4	http://www.canglou.com/manual/php_m.../index.html	744
5	http://www.cnk8.com/book/php/index.html	580

Table 2 Top 5 pages linked by anchor text “起点” sorted by number of sites

Rank	URL	#Sites
1	http://www.cmfu.com/	709
2	http://www.cmfu.com/index.asp	28
3	http://www.chinabyte.com/key/4842/159842.html	6
4	http://bbsx.cmfu.com/bbsx/index.asp	3
5	http://lyz7707.blogchina.com/3863865.html	2

other miscellaneous websites have low in-sites numbers. Hence site-distribution for this query is more reasonable than link-distribution. We use #sites(q, p) to denote the number of sites which link to the Web page p with anchor text q. For each Web page p in L_q, we define the percentage of sites for page p as:

$$P_{\text{site}}(p\mid q)=\frac{\#\text{Sites}(q,p)}{\#\text{Sites}(q)}=\frac{\#\text{Sites}(q,p)}{\sum_{i\in L_q}\#\text{Sites}(q,i)}$$

Corresponding site entropy (**SiteEntropy**) of query q is defined as the following.

$$\text{SiteEntropy}(q)=-\sum_{p\in L_q}P_{\text{site}}(p\mid q)\log P_{\text{site}}(p\mid q)$$

2 Dataset

In this section, we first introduce the benchmark queries, then introduce the click-through data and anchor data which will be used in our experiments.

2.1 Benchmark Queries

We create a benchmark query set for user goal identification task. Our benchmark queries are composed of three different parts: (1)116 site names selected from a widely-used Chinese web directory site hao123.com. Most of these site names are navigational queries. (2) 68 queries generated based upon the test set used in a Chinese search engine contest. The queries are manually selected keywords in the descriptions of the topic distillation tasks[2]. Most of these queries are informational queries. (3) 100 queries selected from the top queries in 2006 given by Baidu.com[3].

Altogether there are 284 queries. After matching with the click-through data and anchor data we use, we find there are 78 queries which lose either anchor data or click-through data. The identification task for these queries is remained as our future work. We use the left 206 ones as our final benchmark queries. There queries are then judged by three annotators. All the three annotators are university students majored in computer science and are familiar with search engine. A query type, either navigational or informational, is assigned to each query after annotators’ discussion.

2.2 Click-through Data

We use a large-scale query log provided by Sogou Labs. This data are downloaded from the web page http://www.sogou.com/labs/dl/q.html. This dataset includes all queries and corresponding clicks in August 2006. It contains 10,812,075 queries and 21,426,131 clicks. There are totally 3,118,907 unique queries and 8,621,580 documents. Query sessions are provided according to cookie information and there are totally 5,130,767 sessions. We extract all click-through data for the benchmark queries and then extract all click-through-related features based on these data.

2.3 Anchor Data

We extract anchor data from a web archive which contains more than 100 million Chinese Web pages crawled in late 2006. We scan all Web pages to identify the anchors that have the same text as our benchmark queries. The Web pages which contain the anchors and those which the anchors link to are then extracted and merged. Link distribution and site distribution are then generated based on these data.

3 Query Type Classification Experiments

In this section, we use our proposed features to train user goal identification models and evaluate the effectiveness of features. We use a support vector machine (SVM) classifier[8] with RBF kernel to accomplish the classification task. We use a 5-fold cross validation in the experiments. More specifically, the original benchmark query set is randomly split into five parts. We use *each* four parts for training and use the left part for test. Experiments results of the five different enumerations are then averaged as final results.

We use four evaluation metrics including precision, recall, F1, and accuracy[9] [9] to judge the effectiveness of the query type identification task. These metrics are calculated separately for two kinds of queries and then are averaged (i.e., we use macro-average).

3.1 Features in Prior Work

To evaluate the performance of our proposed features, we compare them with the features proposed in previous work. We implement five prior features which are most effective to our best knowledge. Three of them are proposed by Lee et al.[3] : median of click distribution (MedianClick), median of anchor-link distribution (MedianLink), and average number of clicks per query session (AvgClick). The other two are proposed by Liu et al.[7] : n clicks satisfied (nCS) and top n results satisfied (nRS). We give some short descriptions for these features in the following. Please refer the original papers if you are interested in the details of these features.

- **MedianClick**: *Median of click distribution*. For a navigational query, the median of click distribution should highly skew toward rank one. While for an informational query, the median of click distribution are relatively larger since users usually click multiple documents.

[2] Original descriptions for queries can be found at: http: //www.cwirf.org/2005WebTrack/05.td_topics.71-145

[3] http://www.baidu.com/2006/06top10.html

- **MedianLink**: *Median of anchor-link distribution.* Similar with MedianClick, for a navigational query, the median of anchor-link distribution should highly skew toward rank one. Considering the problem of link spamming we discussed above, we also implement a corresponding site version of MedianLink. We name it **MedianSite**.
- **AvgClick**: *Average number of clicks per query*. Intuitively, for a navigational query, the user is most likely to click on only one or a few results that correspond to the Website the user has in mind.
- ***n*CS**: *The percentage of sessions of the query that involves less than n clicks*. Navigational type queries have larger nCS values than informational/transactional ones. Following the authors' choice, we set n=2 for this feature.
- ***n*RS**: *The percentage of sessions of the query that involves clicks only on top n results*. Navigational type queries have larger nRS than informational/transactional ones. Following the authors' choice, we let n=5 for this feature.

Altogether we implement 10 features, including 4 features we propose and 6 features derived from prior work.

3.2 Evaluation of Individual Features

After generating all 10 features for our benchmark queries, we first experiment with each single feature to evaluate their effectiveness for user goal identification. Table 3 summarizes experimental results. We find the following conclusions from this table:

First, our entropy-based features SiteEntropy and ClickEntropy perform very well. Furthermore, feature SiteEntropy is the most one among all features. It yields about 94% accuracy.

Second, when using anchor data, entropy-based features (LinkEntropy and SiteEntropy) are more effective than median-based features (MedianLink and MedianSite). When using click-through data, median-based feature MedianClick is better than entropy-based feature ClickEntropy. We think it is because that entropy is less stable than median when the noise we introduced above exists. After removing some noise, we find that the entropy-based feature DomainClickEntropy is better than MedianClick.

Third, features nCS and AvgClick perform less well. These features assume that users click many Web pages for an informational query. By observing the data, we find that when users issue an informational/transactional query, they may also just click one or a few results. For example, for the query "genetic algorithm", more than 85% users click only less than 2 documents. Users indeed want to use this query to learn about genetic algorithm, but they may already get what they want just after they click one or two documents. For this reason, features nCS and AvgClick are less effective than expected.

Fourth, SiteEntropy and MedianSite perform significantly better than LinkEntropy and MedianLink. This proves that the number of sites is more reliable than the number of links when using anchor data.

Table 3 Query type identification experimental results when using each single feature

Features	Precision	Recall	F1	Accuracy
LinkEntropy	0.7300	0.7809	0.7839	0.8014
SiteEntropy	**0.9281**	**0.9388**	**0.9333**	**0.9368**
ClickEntropy	0.8516	0.8411	0.8462	0.8599
DomainClickEntropy	0.9028	0.8878	0.8952	0.9038
MedianLink	0.7597	0.7690	0.7641	0.7767
MedianSite	0.9013	0.9299	0.9154	0.9125
MedianClick	0.8733	0.8858	0.8795	0.8847
AvgClick	0.6369	0.5212	0.5666	0.6411
nCS	0.5696	0.5257	0.5372	0.6459
nRS	0.8526	0.8561	0.8542	0.8653

3.3 Combination of Multiple Features

We then split all 10 features into three groups. The first group includes the features proposed by Lee et al.[3] and feature MedianSite. The second group contains feature nCS and nRS proposed by Liu et al.[7] . Our entropy-based features are categorized into the third group. We use each group to identify user goals and report experimental results in Table 4. This table shows that our entropy-based features are much better than others. These features yield about 8% accuracy improvement over other groups. We also find features proposed by Lee et al.[3] are better than those proposed by Liu et al.[7] for our benchmark queries. Furthermore, we find the combination of four entropy-based features also outperforms each single feature in this group.

Table 4 Performance of multiple features

Features	Pre.	Rec.	F1	Acc.
MedianLink+MedianSite+MedianClick+AvgClick	0.8604	0.8788	0.8695	0.8749
nCS+nRS	0.8526	0.8561	0.8542	0.8653
LinkEntropy+SiteEntropy+ClickEntropy+DomainClickEntropy	**0.9475**	**0.9541**	**0.9507**	**0.9520**

We further group 10 features into anchor-based features (G1) and click-through-based features (G2) by data source of feature. Click-through-based features are further divided into distribution-based features (G2.2) and session-based features (G2.1). Distribution-based features are extracted based on URL/domain click distribution for a query, while session-based features are extracted based on the statistics of query sessions. Table 5 shows experimental results by using each group of features. We find anchor-based features perform better than click-through-based features. It may be because that anchor text is more clean and accurate than user clicks. Furthermore, in click-through-based features, the distribution-based ones are better than click-based ones. As we described above, some informational queries may also have small click numbers which may reduce the accuracy of session-based features.

Table 5 Performance of different types of features

Group	Features	Pre.	Rec.	F1	Acc.
G1	LinkEntropy+SiteEntropy +MedianLink+MedianSite	**0.9345**	**0.9426**	**0.9384**	**0.9418**
G2.1	nCS+nRS+AvgClick	0.8538	0.8531	0.8533	0.8654
G2.2	MedianClick+ClickEntropy+DomainClickEntropy	0.8907	0.8876	0.8891	0.8989
G2	G3+G4	0.9104	0.8997	0.9047	0.9136

3.4 Best Combination of Features

We experiment with all 10 features and give the results in Table 6. We find using all 10 features could improve classification accuracy but the improvement is not significant. It may be because that our entropy-based features have enough information coverage already. We use the forward search feature selection algorithm (SFFS)[10] to select a group of features which yield best performance. We find the combination of only two features SiteEntropy and MedianClick performs better than using all 10 features. They yield **97%** classification accuracy. Interestingly, in term of data source, there is an anchor-based feature and a click-through-based one. In term of feature type, there is an entropy-based feature and a median-based one. Feature DomainClickEntropy, which performs better than MedianClick, is excluded from the features. These results tell us that the diversity of features is very important to get optimal classification accuracy.

Table 6 Performance of all and best features

Group	Features	Pre.	Rec.	F1	Acc.
All	All 10 features	0.9540	0.9519	0.9529	0.9570
Best	SiteEntropy+MedianClick	**0.9726**	**0.9756**	**0.9741**	**0.9761**

4 Conclusions

In this paper, we proposed four novel *entropy-based* features extracted from anchor data and click-through data for user goal identification task. We used a SVM classifier to classify queries based on these features. The comparison of experimental result showed that our proposed entropy-based features are more effective than prior ones. They yielded about 8% accuracy improvement. Furthermore, by combining multiple features we can correctly identify the goals for more than 97% benchmark queries. Besides these, we also got several important conclusions about user goal identification: First, anchor-based features are more effective than click-through-based features; Second, number of sites is more reliable than number of links; Third, click distribution-based features are more effective than query session-based ones.

Since all the features we used in this paper are extracted based on anchor and click-through, they are not available for the tail queries which lack corresponding anchor and click information. We will continue our work to solve this problem in future work.

References

[1] Broder A. A Taxonomy of Web Search[J]. *SIGIR Forum*, 2002, **36**(2): 3-10.

[2] Rose D. E. and Levinson D. Understanding User Goals in Web Search[C]//*Proceedings of the 13th international Conference on World Wide Web*. New York: ACM Press, 2004: 13-19.

[3] Lee U., Liu Z., and Cho J. Automatic Identification of User Goals in Web Search[C]//*Proceedings of the 14th international Conference on World Wide Web*. New York: ACM Press, 2005: 391-400.

[4] Craswell N, Hawking D and Robertson S. Effective Site Finding using Link Anchor Information[C]//*Proceedings of ACM SIGIR '01*. New York: ACM Press, 2001:250-257.

[5] Westerveld T, Kraaij W, and Hiemstra D. Retrieving Web Pages Using Content, Links, URLs and Anchors[C]// *Proceedings of the Tenth Text REtrieval Conference (TREC-10)*. Gaithersburg: National Institute of Standards and Technology, 2001:663-672.

[6] Kang I and Kim G. Query Type Classification for Web Document Retrieval[C]//*Proceedings of the 26th Annual international ACM SIGIR Conference on Research and Development in informaion Retrieval*. New York: ACM Press, 2003: 64-71.

[7] Liu Yiqun, Zhang Min, Ru Liyun, and Ma Shaoping. Automatic Query Type Identification Based on Click Through Information[J]. *Lecture Notes in Computer Science*, 2006, **4182**: 593-600.

[8] Vapnik V. Principles of Risk Minimization for Learning Theory[J]. *Advances in Neural Information Processing Systems*, 1992, **3**:831-838.

[9] Yang Y. An Evaluation of Statistical Approaches to Text Categorization[J]. *Information Retrieval*, 1999, **1**(1-2): 69-90.

[10] Jain A.K. and Zongker D. Feature-Selection: Evaluation, Application, and Small Sample Performance[J]. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 1997, **19**(2):153-158.