International Journal of Computer and Communication Technology

Volume 4 | Issue 3 Article 8

July 2013

Web Search using Improved Concept Based Query Refinement

Ralla Suresh
Pulipati Prasad Engineering College,Khammam, A.P, rella.suresh@gmail.com

Saritha Vemuri
Pulipati Prasad Engineering College,Khammam, A.P, sarithak.vemuri@gmail.com

Swetha V
M.Tech, JNTU Hyderabad,AP, India., jammula.swethareddyi@gmail.com

Web Search using Improved Concept Based Query Refinement

Ralla Suresh[1],Saritha Vemuri[2],Swetha V[3]

[1] ***Research Scholar, Rayalaseema University, AP, India.***
[2,3] ***M.Tech, JNTU Hyderabad,AP, India.***
E-mail: rella.suresh@gmail.com, jammula.swethareddyi@gmail.com, sarithak.vemuri@gmail.com

---

Abstract: The information extracted from Web pages can be used for effective query expansion. The aspect needed to improve accuracy of web search engines is the inclusion of metadata, not only to analyze Web content, but also to interpret. With the Web of today being unstructured and semantically heterogeneous, keyword-based queries are likely to miss important results. . Using data mining methods, our system derives dependency rules and applies them to concept-based queries. This paper presents a novel approach for query expansion that applies dependence rules mined from a large Web World, combining several existing techniques for data extraction and mining, to integrate the system into COMPACT, our prototype implementation of a concept-based search engine.

Keywords- Accuracy, Metadata, concept based, semantic, Refinement.

---

i. Introduction

One important aspect needed to improve the accuracy of web search engines is the inclusion of metadata, not only to analyze Web content, but also to interpret and expand user queries. Often regularities that can be exploited are contained in the original data itself. These regularities, e.g. correlations, can be seen as gainful metadata as these regularities can be leveraged for query refinement and disambiguation. Until then, the vast majority of Web pages will still be plain HTML without any semantic annotations. Even though some Web sites have gradually moved their content to XML, it is often no schematic and exposes wide diversity in terms of document structures and tag names. External ontologies (in contrast to integrated met information in the Semantic Web) could help to interpret no annotated semi-structured information, but they are either too specialized, e.g. in the area of bio informatics, or too broad, like the general-purpose thesaurus WorldNet. Additionally, hardly any existing ontology contains instance or property information , and finding reasonable quantitative similarity measures for related concepts in an ontology is a difficult problem. Therefore the only way to automatically acquire and maintain Meta information is to extract it from existing, non-annotated HTML pages of today's Web.

Much work has been spent so far to recognize the intended structure of HTML pages to extract the contained information. We focus on the most structured and therefore most gainful parts of HTML elements, namely tables and forms.

a) Related Work

Much effort has been spent in the area of semi-automatic or automatic extraction of structured HTML-elements to deduce the intended semantic structure. In this work we adopted and combined especially the work for automatic classification of tables described in and a generic approach for table recognition, described. As this is not the main focus of this work but rather a tool, it could be easily replaced by other approaches. We use information obtained by table extraction as a basis for data mining tasks. Therefore our work is highly related to Data Mining in general, especially association rule mining and to Web Mining, as we use these techniques in a Web based setting. Furthermore as we also want to use information contained

in HTML forms, this work is related to many works that Finally query refinement is a goal of this work that already has been addressed by many papers, especially query expansion using Word Net. Although much work has been done
in these different areas, we are not aware of works that combine these means in a setting comparable to this work.

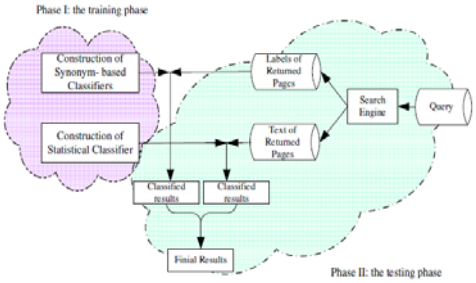
b) Contributions

In this paper we show how to utilize automatically extracted Meta information for

a) The refinement of queries to improve the result quality in terms of recall an precision.
b) The transformation of queries into concept-based queries that can be fed into Web portal forms.
c) The statistical meta data needed for this process is collected during crawling. Our algorithm for query refinement and form matching is surprisingly simple and efficient.

ii. Refinement Analysis

Query Enrichment for Web-query Classification



The query classification problem is not as well-formed as other classification problems such as text classification. The difficulties include short and ambiguous queries and the lack of training data. In this section, we give a formal definition of the query classification problem.

Query Classification:

The aim of query classification is to classify a user query Qi into a ranked list of n categories Li1, Li2, : : :, Lin, among a set of N categories fL1, L2, : : :, LN. Among the output, Li1 is ranked higher than Li2, and Li2 is higher addressed this problem. than Li3, and so on. The queries are collected from real search engines submitted by Web users. The
meanings and intension of the queries are subjective.* The target categories are a tree with each node representing a category. The semantic meanings of each category are defined by the labels along the path from the root to the corresponding node.

PHASE I: CLASSIFIER TRAINING

We now discuss phase I of our approach in detail. In this phase, we train classifiers for mapping from intermediate objects into the target categories. As noted, a main problem here is the lack of training data, a difficulty which makes many previous machine learning methods unable to be applied. The objective in phase I is to collect the data from the Web that can be used to train classification functions.

PHASE II: QUERY CLASSIFICATION

Phase I of our algorithm is designed to collect data for training the mapping functions which include the synonym-based and statistics-based classifiers. Phase II of our algorithm is devoted to classifying a query to one or more target categories based on the classifiers.

iii. Query Enrichment

Query enrichment is a key step because our goal is to classify short and ambiguous queries without any additional descriptions about these queries. After this step, two kinds of information for each query are collected. One is the list of Web pages related to the target query. The other is the set of categories corresponding to the related pages. These two kinds of information will be leveraged by the two kinds of classifiers trained in phase I respectively.

In our approach, we send each query into multiple search engines which can provide options for both directory search and Web search. Directory search in a search engine refers to search algorithms that return the related pages of a query together with the pages' categories. Since these categories of Web pages are labeled
by human labelers, it is appropriate to use them to classify the queries.

However, not all the pages indexed by the search algorithm contain category information; in this case, Web search can return more related pages than

Directory search. Based on the contents of the pages returned by Web search, we can classify the queries using a text classification algorithm.

In summary, to enrich a query through search engines, we use the following 2 steps:
(1) we first try to get the related pages through Directory Search";
(2) if we cannot get any results from step 1, we try Web Search";

iv. Pre-processing Metadata

Before we can actually analyze the Meta data we have collected in the previous step, we have to clean it from any artifacts introduced by errors during collecting. Such errors may occur when extracting the property descriptors as well as during the extraction of instance data. To fill missing property labels, we first consider all instances derived from the same Web site together, assuming that all extracted tables have the same structure and hence all extracted instances have the same set of columns c1, ..., cn without label. For each unlabeled column ci, we compute the distribution pi of its values and compare it to the distributions qj of values in all labeled properties l1, ..., lm in our complete database, using the well known Kullbach-Leibler divergence of two value distributions:KL(pi||qj) =X x 2ci pi(x)log pi(x) qj(x) If we find at least one candidate for which the Kullbach- Leibler divergence is smaller than a predefined threshold, we label the column with the candidate lj with the lowest. Otherwise the unlabeled column is completely removed from the database. Our algorithm also makes some effort to preprocess values, e.g., by cutting additional characters and normalizing values in different formats or with different units.

However, as this information integration problem is difficult to solve exactly, we limit ourselves to simple heuristics here. As a result we get a table with a schema comprising all attributes that we have found so far, where each column corresponds to a property. This table contains a lot of columns from many domains, but each single row (which is an instance from a single domain) has non-null entries in only a few of them. For this reason it is possible to determine clusters of rows that share many properties (not values), yielding a separation of rows from different application domains, because instances of completely different domains (like cars and books) typically won't share a lot of properties or maybe no property (like make, model etc. vs. title, author etc.) at all.

Table 3. Data Sets

PROPERTY	VALUE	RECKEY	P_Bucket	V_Bucket	Partition
make	audi	17	1	17	1
model	tt	17	2	7	1
model	1.8	17	2	6	1
mileage	5133	17	6	113	1
transmission	manual	17	7	1	1
trade name	aspirin	18	21	3	2
generic drug	acetylslicylic acid	18	22	7	2
make	audi	18	1	17	1
model	a6	18	2	2	1
model	tdi	18	2	4	1

Figure 4. Preprocessed Data

Figure 4 shows the database from Figure 2 after preprocessing. The different properties and values are mapped onto numerical values (V bucket and P bucket); the column Partition indicates the cluster of the record.

v. Experiments

a) Setup

The system was run on a dedicated PC (Dual-Intel 3 GHz, 2 GB RAM) running Windows 2003 Server. Our software is implemented in Java (1.4.1) using the WEKA 3 Library for data mining. The data was stored in an Oracle10g database running on the same machine. For collecting our source data we used the BINGO! focused Crawler.

b) Query Refinement

We made preliminary experiments on the two application domains pre owned car advertisements and drugs. More comprehensive experiments are subject of future work. We split each of our domains into training and a testing set. On the training sets we tried to discover association rules 1.

Cars drugs

	Cars	Drugs
Pages	31250	121
Records	37672	484
Rules	7841	311

Table 3 shows the number of pages that that contained genuine tables

(according to our classifier) and have been used for extraction. The number of pages of the drug domain is relatively small as most pages in this domain do not contain any HTML table or the contained tables could not be automatically extracted.

For the following queries Q1 and Q2 we show the steps of our algorithm in detail.

Q1 (domain car ads): Audi A6 diesel
Q2 (domain drugs): Aspirin
The queries were mapped onto the following concept-value pairs:

For all 20 queries we measured precision among the first 10 results and recall. To determine the recall we intensely analyzed our source data. Figure 5 shows recall and precision of the queries Q1 and Q2 in comparison to the expanded queries Q1" and Q2". Although the precision of Q1" is slightly worse compared to the original query Q1, the expanded queries outperformed the original queries. Using Rule R2 1 we not only get pages about Aspirin but also about ASS Ratiopharm that is a drug of the same composition as Aspirin 2. Figure 6 shows the improvements concerning recall and precision of the refined queries (marked with"). Again the 2 this drug is only available in Germany

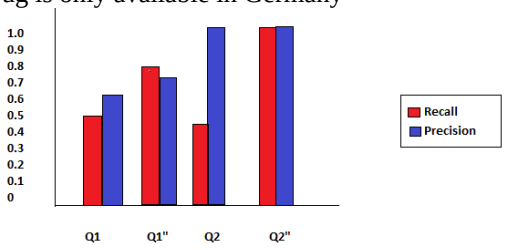


Figure5.a) Recall and precision 1

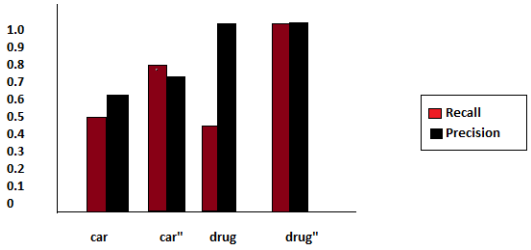


Figure 5. b) Recall and Precision 2

vi. Conclusion and Future Work

This paper has successfully shown that information extracted from Web pages can be
used for effective query expansion. We plan to integrate the system presented in this paper into COMPACT, our prototype implementation of a
Concept-based search engine. Using data mining methods, our system derives dependency rules
and applies them to concept-based queries. Even though we presented only preliminary experiments from two simple application domains, the results are promising and pave the way for future research in this area. In this, we want to implement large-scale experiments with data from different application domains to prove that our approach can be applied generally.

References

1. J. Graupmann, M. Biwer, C. Zimmer, P. Zimmer, M. Bender,M. Theobald, G. Weikum. COMPASS: A Concept-based Web Search Engine for HTML, XML, and Deep Web Data., 2004.

2. R. Baumgartner, S. Flesca, G. Gottlob: Visual Web Information Extraction, 2001

3. V.Crescenzi, G. Mecca, P. Merialdo: RoadRunner: Automatic Data Extraction from Data-Intensive Web Sites, 2002.

4. F. Ciravegna, A. Dingli, D. Guthrie, and Y.Wilks. Integrating information to bootstrap information extraction from web sites., 2003.

5. Y .Wang, J. Hu. A machine learning based approach for table detection on the web, *2002*.

6. S.J. Lim, Y.-K. Ng. An automated approach for retrieving hierarchical data from HTML tables. In *CIKM 1999*, pages:, 1999.

7. M. Yoshida, K. Torisawa and J. Tsujii. A method to integrate tables of the World Wide Web In Proceedings of the International Workshop on Web Document Analysis (WDA 2001),

8. S.Brin. Extracting patterns and relations from the World-Wide Web. In *WebDB 1999*.

9. P. D. Turney. Mining the Web for synonyms: PMI-IR versus LSA on TOEFL. In Proceedings of the Twelfth European Conference on Machine Learning, 2001.