INFORMATION EXTRACTION FROM TEMPLATE-GENERATED HIDDEN WEB DOCUMENTS

Yih-Ling Hedley, Muhammad Younas, Anne James
School of Mathematical and Information Sciences, Coventry University
Priory Street, Coventry CV1 5FB, UK
{y.hedley, m.younas, a.james}@coventry.ac.uk

Mark Sanderson
Department of Information Studies, University of Sheffield
Regent Court, 211 Portobello St, Sheffield, S1 4DP, UK
m.sanderson@shef.ac.uk

ABSTRACT

The larger amount of information on the Web is stored in document databases and is not indexed by general-purpose search engines (such as Google and Yahoo). Databases dynamically generate a list of documents in response to a user query – which are referred to as Hidden Web databases. Such documents are typically presented to users as template-generated Web pages. This paper presents a new approach that identifies Web page templates in order to extract query-related information from documents. We propose two forms of representation to analyse the content of a document – Text with Immediate Adjacent Tag Segments (TIATS) and Text with Neighbouring Adjacent Tag Segments (TNATS). Our techniques exploit tag structures that surround the textual contents of documents in order to detect Web page templates thereby extracting query-related information. Experimental results demonstrate that TNATS detects Web page templates most effectively and extracts information with high recall and precision.

KEYWORDS

Hidden Web Databases, Information Extraction.

1. INTRODUCTION

Hidden Web databases or searchable databases (Gravano, et al, 2003; Bergman, 2000) maintain a collection of documents such as archives, manuals and news articles. These databases dynamically generate a list of documents in response to users' queries. Consequently, information contained in documents is beyond the indexing capability of general-purpose search engines such as Google and Yahoo – which index Web pages through hyperlinks. In recent years, general-purpose or specialised search engines provide services for the search of information on the Hidden Web. As the number of databases increases, it has become prohibitive for search services to evaluate individual databases in order to answer users' queries.

Current research studies focus on the techniques of database selection (Callan and Connell, 2001; Lin and Chen, 2002; Sugiura and Etzioni, 2000) or database categorisation (Meng et al, 2002) in an attempt to automate information searches. In particular, statistical information (such as terms and frequencies) is collected from database sources for their selection or categorisation. However, in the domain of Hidden Web databases, such statistics are often unavailable. In practice, it is not feasible to retrieve all documents from databases to gather their terms and frequencies. Therefore, a number of techniques sample database documents and automatically generate terms and frequencies in order to discover their contents (Callan and Connell, 2001; Lin and Chen, 2002; Sugiura and Etzioni, 2000). However, these techniques often extract terms that are irrelevant to queries since a number of terms retrieved are used in Web page templates for descriptive or navigation purposes.

There exist a number of techniques that extract information from dynamically generated Web pages. For instance, Rahardjo and Yap (1999) adopt approximate string matching techniques for information extraction

from Web pages, but their approach is limited to textual contents only. In contrast, Caverlee et al (2003) discover dynamically generated objects from Web pages by analysing their tags and texts in tree-like structures. However, this approach requires Web pages that contain well-conformed HTML tag trees. Moreover, Web pages are clustered into groups of similarly structured Web pages based on a set of pre-defined page templates, such as exception page templates and result page templates.

In this paper we propose a new approach that identifies the sections of template-generated Web documents that are relevant to queries. Our approach exploits both textual contents and tag structures – particularly the tags that surround a text segment. These include: (i) immediate adjacent tag structures surrounding a text segment and (ii) a list of tag structures which surround a text segment.

A number of experiments have been conducted to assess the effectiveness of the proposed approach. The results show that our techniques provide an effective mechanism to detect Web page templates thereby extracting query-related information. Contributions of this approach include: (i) the effective detection of Web page templates and extraction of query-related information (ii) the generation of terms and frequencies with a higher degree of accuracy. Statistics of improved accuracy can then enhance the effectiveness of database categorisation.

The paper is organised as follows. Section 2 describes current approaches to collecting statistics about databases and problems associated with information extraction. It also summarises existing techniques that extract information from Web pages and dynamically generated documents. Section 3 presents the proposed approach. Experimental results are discussed in section 4. Section 5 concludes the paper and provides direction for further research.

2. RELATED WORK

Recent research discovers the content of Hidden Web databases through sampling their documents (Callan and Connell, 2001; Lin and Chen, 2002; Sugiura and Etzioni, 2000). The terms and statistical information generated from sample documents is referred to as 'Language Models' (Callan and Connell, 2001), 'Textual Models' (Lin and Chen, 2002; Sugiura and Etzioni, 2000) or 'Centroids' (Meng et al, 2002). Such statistical information is then utilised in the process of database selection (Callan and Connell, 2001; Lin and Chen, 2002; Sugiura and Etzioni, 2000) or database categorisation (Meng et al, 2002). However, these techniques extract terms that are generally used in Web page templates for descriptive or navigation purposes. For instance, Callan and Connell (2001) sample documents from Combined Health Information Database (CHID) and the language model generated consists of terms (such as 'Author' and 'Home') with high frequencies, which are not relevant to its content. It is proposed that additional stop-word lists can be used to eliminate irrelevant terms, but such a technique can be difficult to apply in practice. Textual models contain additional topic terms through sampling Web databases. However, these models also contain terms that are found in Web page templates and thus irrelevant to database contents.

Current techniques employed to extract information from Web pages are of varying degrees of complexity. For instance, Rahardjo and Yap (1999) focus on textual contents of Web pages only, and applies approximate string matching techniques to extract texts that are different. This approach is limited to finding textual similarities and differences. It might not be sufficient to extract query-related information from Web pages that are dynamically generated using different templates. The approach proposed in (Caverlee et al, 2003) discovers objects from dynamically generated Web pages by analysing tag structures and textual contents. It requires that Web pages contain well-conformed HTML tag-trees - otherwise additional computation is needed to convert Web pages into appropriate tag-tree structures. Moreover, this approach clusters Web pages into groups of similarly structured Web pages based on a number of pre-defined page templates, such as exception page templates and result page templates.

In contrast, our approach considers texts and adjacent tag structures as opposed to analysing document contents based on text only or tree-like structures. In addition, we detect Web page templates used to dynamically generate documents from databases. This is different from the approach that analyses Web pages, which are grouped according to a pre-defined set of page templates.

3. EXTRACTION OF QUERY-RELATED INFORMATION

This section describes the extraction of information relevant to queries from Hidden Web documents - which we refer to as *query-related information*. Our paper proposes an approach that analyses textual contents and adjacent tag structures to identify Web page templates employed to generate documents. The identification of templates facilitates information extraction from documents. Moreover, it improves the accuracy of terms and frequencies generated from documents. Figure 1 diagrammatically shows the processes of extracting query-related information from documents sampled from Hidden Web databases.

Our technique first samples documents from a given database with keywords. Query keywords can be obtained by randomly selecting terms from frequently used words or those contained in documents retrieved from the database, as proposed by Callan and Connell (2001). The contents of each document are converted into a list of text and tag segments. The document is then represented by a set of text segments along with their adjacent tag segments. Adjacent tag structures of a text segment describe how the text is presented in a Web document and how the text is structured in relation to neighbouring texts. We propose two forms of representation regarding texts and tags contained in a document. These include immediate adjacent tag structures associated with a given text segment and a list of tag segments which surround the text segment.

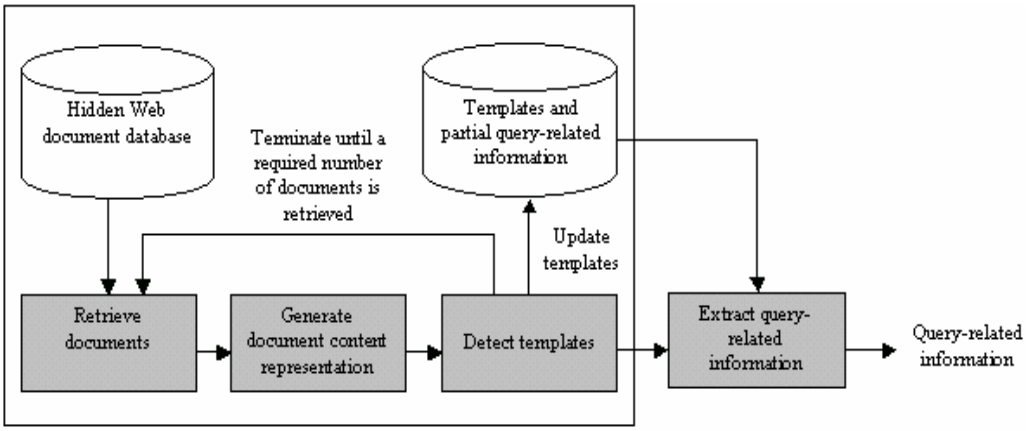

Figure 1. The architecture overview for extracting query-related information from Hidden Web documents

The sampled documents are analysed based on the proposed content representation in order to identify patterns. Information about a template is stored when identical patterns are found in the first document and a subsequently retrieved document. The contents of the two documents (excluding identical patterns) are extracted and assigned to the group associated with the template. If no identical patterns are found, both documents are stored for detecting future templates. Subsequent sampled documents are compared with the template(s) and documents in which no patterns have been found.

When a required number of documents are sampled, information about one or more templates is stored. Two or more documents are associated with each template from which the documents are generated. Text similarity is then computed for the text segments (with identical adjacent tag structures) contained in different documents generated from a given template to determine their similarities. The text segments that exceed a similarity threshold are eliminated. Finally, the textual contents of remaining text segments from the documents are extracted.

Section 3.1 describes the proposed techniques to represent the content of a document. The template detection mechanism and the processes to extract query-related information are detailed in sections 3.2 and 3.3 respectively.

3.1 Document Content Representation

Documents retrieved from Hidden Web databases are first converted into a list of (HTML) tag segments and text segments. Tag segments include starting tags, ending tags or single tags. A text segment is the text that

exists between two tag segments. This paper identifies each text segment through its textual content and the adjacent tag segments.

Two forms of content representation regarding tag and text segments are proposed. These include Text with Immediate Adjacent Tag Segments (TIATS) and Text with Neighbouring Adjacent Tag Segments (TNATS). Documents are then analysed through the proposed content representation in order to identify and detect Web page templates.

3.1.1 Text with Immediate Adjacent Tag Segments (TIATS)

TIATS represents each text segment by its textual content along with immediate adjacent tag segments. The immediate adjacent tag segments of a text segment are defined as the tags that are immediately located before and after the text segment. Assume that a document contains *n* segments, a text segment, *TxS*, is defined as:

$$TxS = (tx_i, tg_j, tg_k)$$

where tx_i is the textual content of *i* text segment, $1 \le i \le n$; tg_j and tg_k are the tags located immediately before and after tx_i respectively, $1 \le j \le n$, $1 \le k \le n$. For example, consider the following segments in a Web document. The text segment 'Links:' (in bold) is represented as ('Links', </TABLE>, <A>).

<TABLE>…</TABLE>
Links:
<A> Documentation </A>
<A> Tutorials </A>
…

3.1.2 Text with Neighbouring Adjacent Tag Segments (TNATS)

TNATS represents each text segment by its textual content along with a list of neighbouring tag segments. Neighbouring tag segments of a text segment are defined as the list of tag segments that are located immediately before and after the text segment until another text segment is found. Assume that a document contains *n* segments, a text segment, *TxS*, is defined as:

$$TxS = (tx_i, tg\text{-}lst_j, tag\text{-}lst_k)$$

where tx_i is the textual content of *i* text segment, $1 \le i \le n$; $tg\text{-}lst_j$ represents *p* tag segments located before tx_i and $tg\text{-}lst_k$ represents *q* tag segments located after tx_i until another text segment is found. $tg\text{-}lst_j = (tg_1, \ldots, tg_p)$, $1 \le j \le p$ and $tg\text{-}lst_k = (tg_1, \ldots, tg_q)$, $1 \le k \le q$. Consider the following segments in a Web document. The text segment 'SYNOPSIS' (in bold) is represented as ('SYNOPSIS', (, <hr>, <h1>, <a>), (, </h1>, <pre>)).

…

<h1><a>NAME</h1>
B::JVM::Jasmin … <hr>
<h1><a>**SYNOPSIS**</h1>
<pre>
use B::JVM::Jasmin; … </pre>
…

Therefore, given a Hidden Web document, *d*, with *n* text segments, the content of *d* is then represented as:

$$Content(d) = \{TxS_1, \ldots, TxS_n\}$$

where TxS_i represents a text segment, $1 \le i \le n$.

3.2 Template Detection

Documents retrieved from Hidden Web databases are often dynamically generated using one or more templates. Templates are typically employed in order to describe document contents and to assist users in navigation. An example of template-generated documents retrieved from the Electronic News database is given in Figure 2. Information contained in this document can be classified into the two following categories:

- Template-Generated Information. This includes information such as navigation panels, search interfaces and advertisements. For example, Figure 2 shows navigation links (i.e., 'Automotive', 'Business') and advertisements (i.e., 'Don't Just Be Sure. Be D&B Sure.'). Our approach aims to identify such information, which is irrelevant to a user query.
- Query-Related Information. This refers to the information retrieved that is relevant to a query (i.e., the news article shown in Figure 3). Our approach aims to extract information that is relevant to users' queries from documents in order to obtain terms and frequencies with improved accuracy.

Figure 2. A template-generated document in response to a query from the Electronic News database

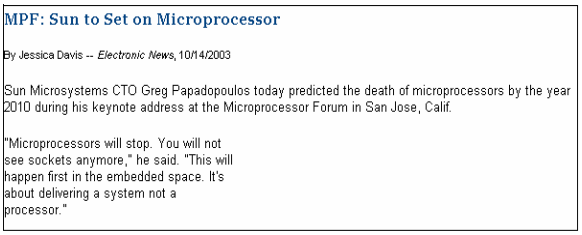

Figure 3. Information that is related to a query

We propose a template detection mechanism that identifies Web page templates used to generate documents in response to a query, which is performed as follows.

(i) The contents of sampled documents are analysed based on their content representations, which are a list of text segments, each with its adjacent tag segments.

(ii) The information of a template is detected and stored if identical patterns (i.e., the matched text segments along with their adjacent tag segments) are found in the first two sampled documents. The documents are assigned into a group associated with the template. Identical patterns are also eliminated from both documents. If no repeated patterns are found, the content representations of both documents are temporarily stored.

(iii) Subsequent templates are detected through comparing the remaining sampled documents with existing template generated or temporarily stored document content representations. Documents that are generated from the same templates are assigned into a group. Any identical patterns are eliminated from documents. Otherwise, their content representations are stored for future template detection.

The process in (iii) is repeated until all sampled documents are analysed. This results in the identification of one or more templates. For each template, there exist two or more documents associated with the template from which the documents are generated. Each of these documents contains text segments that are not found in their respective template. These text segments are partially related to their queries. In addition to the templates generated, the result may contain zero or more documents in which no matched patterns are found.

3.3 Query-Related Information Extraction

Text segments that remain in the sampled documents (as described in section 3.2) are further analysed by computing text similarity. That is, the text segments of different documents from the group associated with a particular template are compared in terms of text similarity. This identifies any text segments with identical tag structure that are similar in their textual contents.

The textual content of a text segment is represented as a vector of terms with weights. A term weight is obtained from the frequencies of the term that appears in the segment. Cosine similarity (Salton and McGill, 1983) is computed on the textual contents of two text segments. The computation of similarity is given as follows:

$$COSINE\ (TxS_i, TxS_j) = \sum_{k=1}^{t} (tw_{ik} \cdot tw_{jk}) / \sqrt{\sum_{k=1}^{t} (tw_{ik})^2} \cdot \sqrt{\sum_{k=1}^{t} (tw_{jk})^2}$$

where TxS_i and TxS_j represent two text segments in a document; tw_{ik} is the weight of term k in TxS_i, and tw_{jk} is the weight of term k in TxS_j.

The similarity is computed for text segments with identical adjacent tag segments only. Two segments are considered to be similar if the similarity of their textual contents exceeds a threshold value. The threshold value is determined experimentally. This process results in the extraction of text segments with different tag structures. It also extracts text segments that have identical adjacent tag structures but are significantly different in their textual content.

4. EXPERIMENTAL RESULTS

Experiments are conducted in order to assess the effectiveness of the proposed techniques in two respects - the detection of templates and extraction of query-related information from Hidden Web documents. Sample documents are analysed based on the proposed content representation, TIATS and TNATS. These are also assessed along with the representation based on text only as employed by Rahardjo and Yap (1999). The text only representation focuses on the textual contents of a document without considering tag structures that format the texts. In contrast, the TIATS and TNATS representation consider the texts and their surrounding tag structures.

Table 1. 8 Hidden Web document databases used in the experiments

Hidden Web databases	URL	Content
help-site	www.help-site.com	Computer user manuals and documentation
devx	www.devx.com	Computing related archives
simplythebest	www.simplythebest.net	IT related articles
techweb	www.techweb.com	IT related articles
wired	www.wired.com	General news articles
electronic news	www.reed-electronics.com	Computing related articles
zdnet	www.itpapers.zdnet.com	IT related articles
chid	www.chid.nih.gov	Health related documents

The experiments are carried out on 8 real-world Hidden Web databases, which provide user manuals, archives and news articles, as shown in Table 1. 10 documents are randomly sampled from each database and a total of 80 documents are retrieved. Sample documents are manually examined to obtain the number of templates used in each database and number of templates that have been detected. Terms extracted using the proposed technique are manually compared with the document from which the terms originate. Recall and precision techniques (of information retrieval systems) are adopted in order to measure the accuracy of query-related information extraction (Salton and McGill, 1983). In this paper, the recall is defined as the ratio of the number of relevant terms retrieved over the total number of relevant terms contained in a document. The precision is given by the ratio of the number of relevant terms retrieved over the total number of terms retrieved from a document.

Experimental results in Table 2 show that TIATS and TNATS detect the number of templates more accurately than the representation based on text only. In particular, TNATS successfully identifies templates

for all databases. This is because TIATS and TNATS both consider the textual content and associated tag structures of a document. For instance, a total of 3 templates are found in 10 documents sampled from the help-site database. The number of templates identified by TIATS and TNATS is 2 and 3 respectively, whereas the number detected based on the text only representation is 1.

Table 2. Number of templates used by the databases and that of detected based on TIATS, TNATS and text only

Hidden Web database	Number of templates			
	Used	Detected		
		TIATS	TNATS	Text only
help-site	3	2	3	1
devx	5	4	5	4
simplythebest	1	1	1	1
techweb	4	4	4	2
wired	1	1	1	1
electronic news	1	1	1	1
zdnet	1	1	1	1
chid	1	1	1	1

Table 3. Recall and precision for the extraction of query-related information from sampled documents of help-site

Document	TIATS		TNATS		Text only	
no	Recall	Precision	Recall	Precision	Recall	Precision
1	1.000	0.947	1.000	0.947	0.981	0.970
2	1.000	0.938	1.000	0.912	0.995	0.946
3	1.000	0.979	1.000	0.961	1.000	0.978
4	1.000	0.979	1.000	0.960	1.000	0.979
5	1.000	0.976	1.000	0.968	1.000	0.976
6	0.999	0.991	1.000	0.985	0.999	0.992
7	0.999	0.989	1.000	0.981	0.999	0.992
8	1.000	0.973	1.000	0.953	1.000	0.979
9	1.000	0.999	1.000	0.994	1.000	0.999
10	1.000	0.954	1.000	0.942	1.000	0.964
Average	1.000	0.973	1.000	0.960	0.997	0.978

Table 3 gives the recall and precision of information extraction from 10 sample documents from the help-site database for the TIATS, TNATS and text only representation. The average of recall for TIATS, TNATS and text only is 1.0, 1.0 and 0.997 respectively. Given that TIATS and TNATS detect Web page templates based on textual contents and tag structures of documents, templates are detected more effectively (as shown in Table 2). As a result, they attain a higher degree of recall, since a number of relevant terms (i.e., non-template terms) are eliminated when the text only representation is employed. That is, the text only representation focuses on the textual content of a document without considering associated tag structures. Consequently, any identical terms or terms that exceed a similarity threshold are eliminated even if these terms are relevant to queries.

Furthermore, the precision attained by TIATS, TNATS and text only is 0.973, 0.960 and 0.978 respectively. The text only representation appears to perform better than TIATS and TNATS in terms of precision. However, our observation is that the text only representation eliminates a larger number of terms (which include template and non-template terms) provided that identical terms or terms that exceed a similarity threshold are found in the sampled documents. Thus, the text only representation attains a higher degree of precision.

By comparison, TIATS and TNATS are applied to analyse textual content and tag structures to determine whether a text segment is associated with templates. However, given that a small set of sampled documents is used in this experiment, they contain insufficient information to detect all of the template terms. Therefore, it appears that TIATS and TNATS extract more template terms than the text only representation. Furthermore, as TNATS requires more information regarding tag structures associated with text segments in order to detect templates, TNATS eliminates template terms less successfully than TIATS. Therefore, the precision attained by TNATS is lower than TIATS. However, it is expected that the greater number of documents sampled, TNATS will achieve a higher degree of precision.

In this paper, we assess the effectiveness of three forms of document content representation in extracting query-related information according to the performance of template detection. Based on this criterion, TNATS extracts query-related information more accurately. The results given in Table 4 demonstrate that TNATS extracts query-related information consistently with high recall and precision. The overall accuracy for 8 databases is 0.97 and 0.95 in terms of recall and precision respectively.

Table 4. Average recall and precision for query-related information extraction using TNATS

Hidden Web databases	TNATS	
	Average recall	Average precision
help-site	1.00	0.96
devx	0.95	0.90
simplythebest	0.92	0.95
techweb	0.99	0.91
wired	0.99	0.98
electronic news	0.97	0.98
zdnet	0.99	0.92
chid	0.94	0.98

5. CONCLUSION

Recent research demonstrates that the contents of text databases can be represented by terms and frequencies retrieved from randomly sampled documents. However, Hidden Web databases dynamically generate documents (such as archives and news articles) using templates. We propose a new approach that analyses documents based on textual contents and their adjacent tag structures. This detects templates and extracts query-related information in order to obtain terms and frequencies with a higher degree of accuracy. Our techniques are in contrast to those that analyse document contents based on text only or in a tree-like structure. Experimental results demonstrate that TIATS and TNATS extract query-related information with a high degree of accuracy in terms of recall and precision. In particular, TNATS is the most effective in the detection of templates.

Future work includes the application of the proposed technique to improve the accuracy of terms and frequencies generated from Hidden Web documents. We will then apply the resultant statistics with improved accuracy to enhance the effectiveness of database categorisation.

REFERENCES

Bergman, M. K. (2000) The Deep Web: Surfacing Hidden Value. Appeared in *The Journal of Electronic Publishing* from the University of Michigan. Retrieved: 10 Feb, 2004, from http://www.press.umich.edu/jep/07-01/bergman.html

Callan, J. and Connell, M. (2001) Query-Based Sampling of Text Databases. *ACM Transactions on Information Systems*, Vol. 19, No. 2, pp 97-130.

Caverlee, J. et al. (2003) Discovering Objects in Dynamically-Generated Web Pages. Technical report, Georgia Institute of Technology.

Gravano, L. et al (2003) QProber: A System for Automatic Classification of Hidden-Web Databases. *ACM Transactions on Information Systems (TOIS)*, Vol. 21, No. 1.

Lin, K.I. and Chen, H. (2002) Automatic Information Discovery from the Invisible Web. *International Conference on Information Technology: Coding and Computing*.

Meng, W. et al (2002) Concept Hierarchy Based Text Database Categorization. *International Journal on Knowledge and Information Systems*, Vol. 4, No. 2, pp 132-150.

Rahardjo, B. and Yap, R. (1999) Automatic Information Extraction from Web Pages. *SIGIR*, pp 430-431.

Salton, G. and McGill, M. (1983) *Introduction to Modern Information Retrieval*. McCraw-Hill, New York, USA.

Sugiura, A. and Etzioni, O. (2000) Query Routing for Web Search Engines: Architecture and Experiment. *Proceedings of 9th International World Wide Web conference*.