Journal of Computer Science 2012, 8 (12), 2053-2061
ISSN 1549-3636

doi:10.3844/jcssp.2012.2053.2061 Published Online 8 (12) 2012 (http://www.thescipub.com/jcs.toc)

Web Document Segmentation Using Frequent Term Sets for Summarization

[1]Chitra Pasupathi, [2]Baskaran Ramachandran and [3]Sarukesi Karunakaran

[1]Department of Information Technology, RMK Engineering College, Tamilnadu, India
[2]Department of Computer Science and Engineering, Anna University, Chennai, India
[3]Hindustan University, Chennai, India

Received 2012-06-04, Revised 2012-06-27; Accepted 2012-12-19

ABSTRACT

Query sensitive summarization aims at extracting the query relevant contents from web documents. Web page segmentation focuses on reducing the run time overhead of the summarization systems by grouping the related contents of a web page into segments. At query time, query relevant segments of the web page are identified and important sentences from these segments are extracted to compose the summary. DOM tree structures of the web documents are utilized to perform the segmentation of the contents. Leaf nodes of DOM tress are merged to form segments according to the statistical and linguistic similarity measure. The proposed system has been evaluated by intrinsic approach making use of user satisfaction index. The performance of the system is compared with summarization without using preprocessed segments. Performance of this system is more promising than the other measures like cosine similarity, jaccard measure which make use of sparse term-frequent vectors, since the most frequent term sets are considered to measure the relevance. Relevant segments alone need to be processed at run time for summarization which reduces the time complexity of the summarization process.

Keywords: Search Engine Optimization, Segmentation, Summarization, Pre-Processing, Query Sensitive

1. INTRODUCTION

The exponential growth of the volume of web documents, poses a hard challenge to the users in locating, retrieving and using the huge contents pooled over WWW. Search engines help the users to search and locate the information to an extent. For each query, the search engine returns thousands of URLs as the result set which includes redundant as well as irrelevant links. Improving the retrieval efficiency to meet the users' personalized need becomes critical in information retrieval domain.

Summarization techniques focus on reducing the time and effort required for the user to understand the core concept of the large by providing a short summary. Query-based summarization technique extracts/abstracts pieces of information from web documents to answer a query. Processing entire document at run time dynamically according to the query is a challenging task for the processing capacity and response time of the automatic summarizers.

Some kind of pre-processing methods like topic based or content based segmentation, sentence clustering can be applied to reduce the processing overhead at run time. This study focuses on content segmentation employing the relevance measurement technique using statistical and linguistic measures.

1.1. Related Works

Related research work can roughly be classified into four major categories of measuring sentence similarity: Word co-occurrence/vector-based document model methods, corpus-based methods, hybrid methods and descriptive feature-based methods.

Chien and Chueh (2012), proposed a system for topic-based hierarchical segmentation model for

Corresponding Author: Chitra Pasupathi, Department of Information Technology, RMK Engineering College, Tamilnadu, India

representation of text streams using two-sided contextual information. The Latent Semantic Analysis (LSA) (Landauer *et al*., 1998; Landauer and Dumais, 1997) and the Hyperspace Analogues to Language (HAL) model (Landauer *et al*., 1998) were two well-known methods in corpus-based similarity. The basic idea of LSA, was that the aggregation of all the word contexts in which a given word did or did not appear would represent the similarity between text units. LSA did not take into account any syntactic information and was thus more appropriate for longer texts.

The HAL (Landauer and Dumais, 1997) method used lexical co-occurrence to produce a high-dimensional semantic space. Similarity between two sentences was calculated using Euclidean distance. Drawback of HAL was due to the building of the memory matrix and its approach to form sentence vectors which did not capture sentence meaning well.

The vector-based document model methods were commonly used in Information Retrieval (IR) systems (Mohamed and Rajasekaran, 2006), where the document most relevant to an input query is determined by representing a document as a word vector and then queries were matched to similar documents in the document database via a similarity metric (Chen and Shen, 2009). An extension of word co-occurrence methods lead to the pattern matching methods which were used in text mining and conversational agents (Iosif and Potamianos, 2010). This technique relied on the assumption that more similar documents would have more words in common. But it is not always the case that texts with similar meaning necessarily share many words (Wang *et al*., 2008).

Semantic Text Similarity (STS) method using the Longest Common Subsequence (LCS) measure for string similarity measure was proposed by Islam and Inkpen (2008). This aproach was improved to compute a similarity measure between text units using feature vectors.

Kogilavani (2012) used Term Synonym Concept Frequency-Inverse Sentence Frequency (TSCF-ISF) to measure the weight of a word to detect dominant concepts in web documents.

Four basic types (Kuppusamy and Aghila, 2012) of web page segmentation method are (1) Fixed length page segmentation (2) DOM based page segmentation (3) Vision based page segmentation (4) Combined/Hybrid method.

VIsion-based Page Segmentation (VIPS) algorithm was proposed by Cai *et al*. (2003) which segmented the web page by simulating the way of human understanding of the web layout structure. This approach used human visual perception model.

Kohlschütter and Nejdl (2008) proposed a segmentation approach which utilized the notion of text-density as a measure to identify the individual segments of the web page by reducing the problem to solve a 1D-partitioning task.

Pnueli *et al*. (2009) described an algorithm that segments a web page recursively to segment the layout of the page and the UI components using the page's rendered image.

Most of these works were not query relevant and the generated segments were not in view of improving summarization process.

The original contributions of this study are:

- Focuses on reducing the time complexity of extractive summarization process at run time
- Relevant pieces of scattered information in the web document are grouped as segments during pre-processing
- DOM tree structures of the web documents are utilized and the relevant leaf nodes are merged to form the segments
- Frequent term sets and the WordNet (wordnet.princeton.edu/wordnet/download/) distance between term sets of the nodes are used to measure the semantic relevance between blocks of text
- The segment details are materialized in relational database and could be used for generating query sensitive summaries on the fly during query time

1.2. Frequent Term Set Based Segmentation and Summary Generation

Query sensitive summarization techniques aim at providing the gist of document with respect to the user query (Mohamed and Rajasekaran, 2006; Chen and Shen, 2009). This short summary is useful in understanding the larger document without reading the entire content. Summarization can be abstractive or extractive in nature. In case of abstractive summarization, NLP techniques are used to generate an abstract of the content by framing sentences. In the later method summary is composed by extracting the important sentences from the document.

Generating summary of the document at run time based on the dynamic query given by the user requires huge processing capacity of the information processing servers. Each sentence in the selected web page need to be verified for relevance to the given query and assigned a score which is a measure of the importance of the sentence. The time required for generating summary can drastically be reduced by reducing or limiting the size of text unit to be processed for summarization.

This study proposes a preprocessing technique in which the relevant pieces of information which are scattered throughout the document can be merged (Chitra *et al*., 2011) into segments. Information about the segments and the keywords are stored in the relational database (Feng *et al*., 2011). At run time the segments relevant to the query having matching keywords alone are identified and processed to generate the summary. This system uses unsupervised technique for segmenting the web documents which can be extended to any domain.

Figure 1 shows the segmentation and summarization process of the proposed system. Web crawler crawls over WWW starting from the seed pages provided and captures and saves the documents in the server's database. Indexer component periodically indexes these documents by creating a keyword based inverse index for the documents.

Indexed documents are segmented using frequent term set based segmentation technique and their segment ID and the frequent terms are saved in segments database (relationalDB) for further usage during summarization.

User enters the query string through the search engine user interface based on which the search engine identifies the set of matching documents from the database and present the URLs to the user according to their rank order. User is required to choose the URL from which he/she wishes to get the gist of the content. Segments relevant to the user query are extracted from Segment Database and processed to generate the summary. The scope for the selection of summary sentences is now reduced to only few segments that are relevant to the query string. This technique is very effective in minimizing the processing overhead of the information servers at run time for dynamic summary generation.

1.3. Frequent Term Sets Based Segmentation

A segment on the web page is the collection of content from the page that is identified as distinct from the rest of the page in some way. **Figure 2** (Frequent term set based segmentation of HTML document) depicts the segmentation (Yen and Hsu, 2009; Chitra *et al*., 2011) using DOM (http://www.w3c.org/DOM/) tree structure. The nodes from left to right of a parent constitute a coherent semantic string of the content (Li *et al*., 2006).

Leaf nodes (Li *et al*., 2006) are considered as micro blocks which are the basic building blocks. Adjacent micro blocks of the same parent tag are merged to form the topic blocks. These topic blocks are stemmed after removing the stop words like a, an, it, to which do not contribute much to the core content of the blocks.

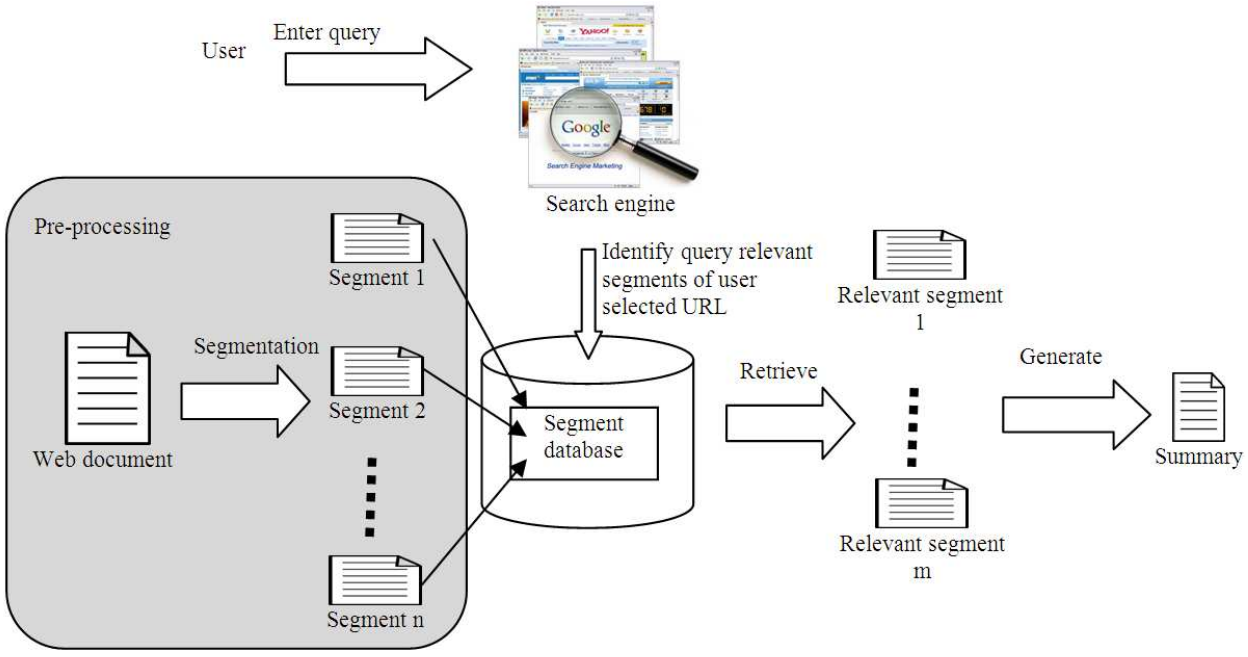

Fig. 1. Segmentation and summary generation

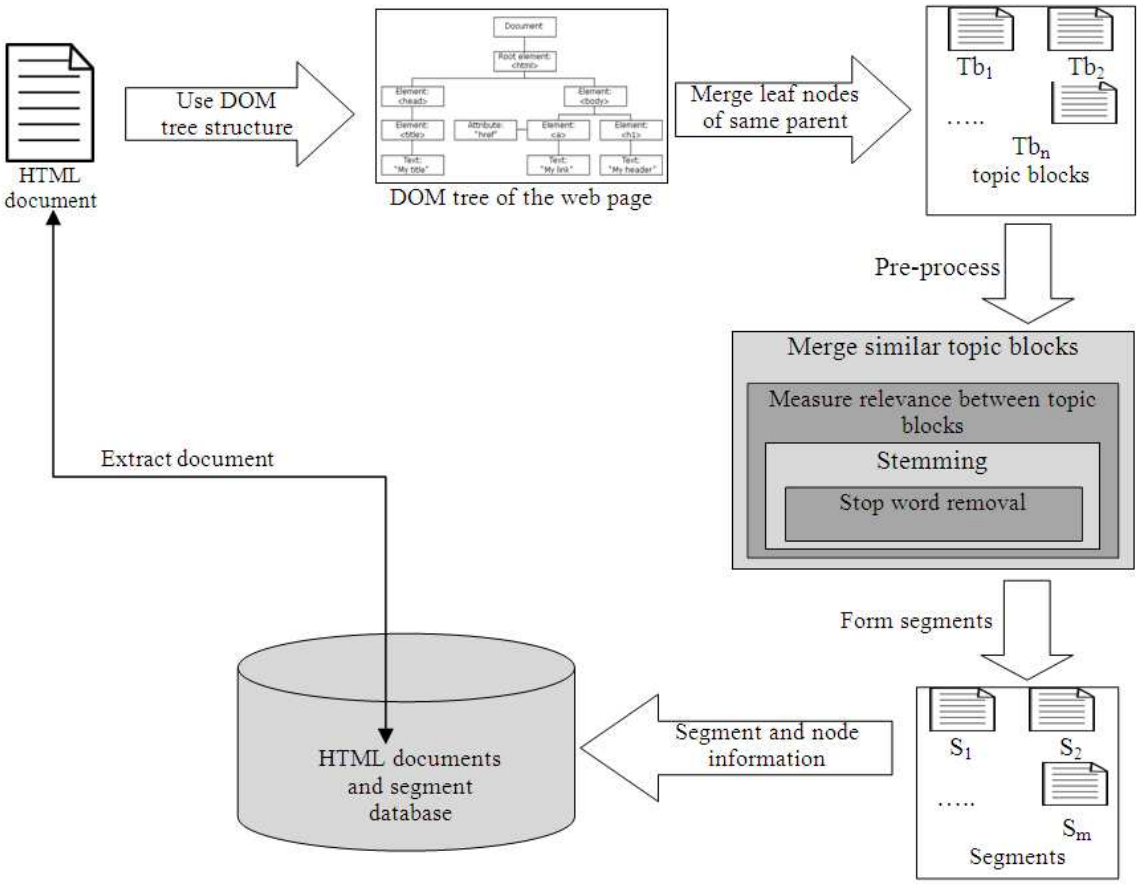

Fig. 2. Frequent term set based segmentation of HTML document

Frequent term sets and their frequency in each of the topic block are identified. Frequent term set based relevance measure is used to measure the semantic relevance between the topic blocks. Topic blocks having similarity above the threshold value α(0.6), are combined to form the concept block. The value of α is chosen so that intra segment relevance is high and inter-segment relevance is less. Relevant topic blocks are expected to have content about the same concept (for example placement and training in a college web site). Similarity measure also considers the WordNet distance between the frequent terms (Pnueli *et al*., 2009) which is considered to be a better measure that simulate human thought proces.

Segmentation is carried out offline for all web documents in the repository and segment details are materialized in relational database for further processing during summarization. The set of sentences in each of these segments are actually present is different parts of the document.

Since DOM nodes are processed, the time taken for processing is less when compared to other vector based and document graph (Wang *et al*., 2008; Mohamed and Rajasekaran, 2006) based models. The processing time required to build the document graph is eliminated in this approach.

1.4. Frequent Term Set Based Segmentation Algorithm

The Frequent Term Sets (FTS) and their WordNet distances are the important factors in measuring the similarity between topic blocks. The segmentation algorithm is described below:

Input: Web document di.
Output: Set of segments $\{S_1,..S_n\}$ of d_i,
Frequent Term sets FTS of d_i, $L=\{t_1,..t_m\}$
FTS of segments S_i, $FTS(S_i) =\{t_{i1},..t_{im}\}$, i=1..n,
number of Segments

Step 1: Mark all leaf nodes as individual micro blocks in the DOM tree.
Step 2: Extend the border of the micro block to include all leaf nodes of the same parent tag to form a topic block so as to have a set of topic blocks TB={tb_1, tb_2, …tb_n}, TB⊂d_i.
Step 3: Get FTS of all topic blocks TB ={tb_1, tb_2, …tb_n} as FTS(tb_i)={$t_{i,1}$,$t_{i,2}$ . . . $t_{i,m}$}, $t_i \subset tb_i$, m: Number of FTS of topic block tb_i
Step 4: The semantic similarity between topic blocks are measured by Equation 1:

$$Sim(tb_1, tb_2) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\left(TF_{(tb1,i)} \times tfweight_{(tb1,i)}\right) + \left(TF_{(tb2,j)} \times tfweight_{(tb2,j)}\right)\right) \times \left(R_{((tb1,i),(tb2,j))}\right)}{\sqrt{\sum_{i=1}^{n}\left(TF_{(tb_1,i)}\right)^2} \times \sqrt{\sum_{j=1}^{m}\left(TF_{(tb_2,j)}\right)^2}} \quad (1)$$

Where:
$TF_{tb1,i}$ = Frequency of i^{th} term in tb_1
$TF_{tb2,j}$ = Frequency of j^{th} term in tb_2
$Tfweight_{(tb1,i)}$, $tfweight_{\{tb2,j)}$ = Weight of term i,j with respect to topic blocks tb_1, tb_2 normalized by the frequency vectors of tb_1, tb_2,calculated (Chitra *et al*., 2011; Islam and Inkpen, 2008; Hao *et al*., 2011) as in Equation 2:

$$tfweight_i = \frac{tf_i}{\sqrt{\sum_{k=1}^{m}\left(tf_k\right)^2}} \quad (2)$$

Where:
$tfweight_i$ = The importance of the i^{th} term with respect to the frequent term vector of i^{th} topic block tbi
tf_i = Frequency of i^{th} term in tbi
$R_{((tb1,i),(tb2,j))}$ = Relevance between i^{th} term in tb_1 and j^{th} term in tb_2 measured using WordNet distance (Yen and Hsu, 2009; Shehata *et al*., 2010) between the terms using Equation 3:

$$R_{((tb1,i),(tb2,j))} = \frac{1}{Distance((t_{b1,i}),(t_{b2,j}))} \quad (3)$$

The similarity score is normalized by the frequency vectors of both topic blocks so that the resulting score lies between the range of 0 to 1.

Step 5: Merge the topic blocks having similarity measure above the predefined threshold α.
Segment S_k={set of topic blocks tb_i}|
$\forall tb_i, tb_j \in S_k$, $sim(tb_i, tb_j) > \alpha$, $tb_i \subset TB$, $tb_j \subset TB$, k=1..n
Step 6: Output Segments S_1, S_2,..S_n and their respective frequent term sets.

Similarity between topic blocks is measured by considering the frequent term sets of topic blocks, tb1 and tb2. Frequency of these terms and their topic block based weight-age are used to measure the similarity score. $tfweight_{(tb1,i)}$ represents the importance of a particular term within the topic block where term frequencies are normalized by the topic block wide frequency factor. WordNet distance (Li *et al*., 2006; Hao *et al*., 2011) between the terms is a useful measure for identifying the semantics relevance between the terms. Words that are semantically closer will get higher score which in turn increases the similarity score between the topic blocks when added to the TF based score.

Segments of all web documents in the repository are identified during pre-processing stage and stored in the database.

1.5. Similarity Metric to Measure Topic Blocks Relevance

Cosine similarity is the most common measure used to find the relevance between text segments (Cai *et al*., 2003, Kumar, 2011). This measure makes use of bags of words approach where the term vector contains more null values. Jaccard measure makes use of frequency of common words to measure the similarity which is very uncommon in real documents. Relevant documents need not contain same set of words to give the same meaning.

Yen and Hsu (2009) and Li *et al*. (2006) have proposed a metric as in Equation 4 to measure the relevance between two documents with respect to which the PageRank of the parent page can be distributed instead of having the PageRank evenly distributed among the outlinks of a page:

$$R(A,B) = \frac{\sum_{h=1}^{n}\sum_{g=1}^{m}\left(TF_{(A,h)} \times TF_{(B,g)} \times R_{(A,h),(B,g)}\right)}{\sum_{h=1}^{n}\sum_{g=1}^{m}\left(TF_{(A,h)} \times TF_{(B,g)}\right)} \quad (4)$$

This measure was tested using dmoz.org (www.dmoz.org/) web pages and proved to be working effectively. Yen and Hsu (2009) considered only the term frequency of all terms in both documents. Importance of particular term within the document is not taken into account which contributes more to measure the relevancy. For one particular term t1, the frequency may be very low in document d1 and very high in document d2.

Table 1. Segment table (table)

DOCID	Segment ID	Node ID	Keywords
Id of Doc 1	S_1	List of node ids	List of keywords in S_1
:	:	:	:
Id of Doc 1	S_{1n}	List of node ids	List of keywords in S_{1n}
:	:	:	:
Id of Doc n	S_n	List of node ids	List of keywords in S_n
:	:	:	:
Id of Doc n	S_{nn}	List of node ids	List of keywords in S_{nn}

Table 2. Keywords index (KWI table)

Keyword	DOCID
Keyword 1	List of Doc ids
Keyword 2	List of Doc ids
Keyword 3	List of Doc ids
:	:
Keyword m	List of Doc ids

Table 3. URL table

DOCID	URL
Doc 1	URL of Doc 1
Doc 2	URL of Doc 2
:	:
Doc n	URL of Doc n

For the same term t1, document 3 and document d4 may be having moderate score compared to d1 and d2. Both will not make any difference in the above mentioned metric. Longer documents are likely to have high frequency for many terms which need to be normalized to find the relevance.

The proposed metric given in Equation 1 considers term frequency, term weight with respect to the topic block and also the WordNet distance to find the relevance between the topic blocks. Hence the segmentation process is more promising than the other aproaches. Consider the previous scenario having four sample documents and a term t1. According to the importance the term weightage changes and also the relavance score is changed. Our metric is more efficient since we consider term frequency, term weight-age and also the WordNet distance. Term weightage itself is normalized by the length of the topic block before multiplying it with the WordNet based relevance. The final score is again normalized by length of the topic blocks as in cosine similarity measure which produces better result than existing measure.

Unlike the vectors used in cosine similarity and Jaccard measures the term frequency vectors of any two topic block contains the words and its frequency. These two vectors need not have the same set of words and in same order (Chitra *et al*., 2011). They can appear in any order as they are obtained after preprocessing the topic blocks. Frequency of each term is added to the frequency of every term in the other vector which is then multiplied by the relevance between these two words as per WordNet Tree structure. The closer terms pair would get higher score as per Equation (1) which may not be similar to each other. For example "college" and "education" are dissimilar but relevant words, would get high score as per Equation (1).

1.6. Materializing the Segments

Information about the identified segments are saved in the relational database (Wang *et al*., 2008). The structure of the segment table and keyword index table are shown below in **Table 1** and 2.

All leaf nodes are numbered from left to right starting from 1 according to DOM tree traversal technique. After segmentation details of the segments are stored on Segment **Table 1** which contains DOCID, SEGID, NODEIDs of all nodes constituting that segment and KEYWORDS present in that segment. Keyword Index Table (KWIT) (**Table 2**) contains an entry for each concept present in the document repository and DOCIDs of all documents containing information about that concept keyword. URL **Table 3** contains mapping from DOCID to actual location of the document.

During query time based on the query keywords, the DOCIDs of relevant documents are identified from Keyword index table. Then the corresponding URLs of these DOCIDs are identified and presented to the user as search result. User now selects a URL to view the summary. From the ST the segments relevant to query keyword are identified and summarization algorithm is applied only on these identified segments.

2. MATERIALS AND METHODS

2.1. Experimentation and Evaluation

Experimentation of the proposed system was conducted using WEBKB (www.cs.cmu.edu/~webkb/) dataset and also real time datasets. WEBKB (Mohamed and Rajasekaran, 2006) dataset contains 6248 web pages

downloaded from four universities containing information related to faculty, students, courses offered, activities, achievements. Real time corpus was built by downloading top 10 web pages from Google search engine for the keywords "Engineering education", "Efficiency of optimization algorithms" and "Global warming". Three diversified domains are chosen to prove that the proposed system is domain independent and unsupervised.

Table 4. Key word index for Real time corpus

Keywords	DOCID
Placement	d1,d4
Training	d1,d4
Activities	d1,d5
Achievements	d1,d7
Optimization Problem	d2,d7,d8
Feasibility Problem	d2,d7
Efficiency	d2,d8
Complexity	d2,d8,d9
Necessary conditions	d2,d10
envelope theorem	d2,d9
Global warming	d3,d11
Observed changes	d3,d12
Green house gases	d3,d12
Solar activity	d3,d13
Feedback	d3,d14
Climate models	d3,d14,d15
Expected environmental effects	d3,d12
Ecological systems	d3,d11

Table 5. Segment table for randomly chosen 3 web documents

DOCID	Segment ID	Node ID	Keywords
d1	S1	2,3,5,7,8	Placement
d1	S2	1,4	Activities
d1	S3	6	Achievements
d1	S4	9	Training
d2	S1	1,2	Algorithms
d2	S2	3,4	Optimization Problem
d2	S3	5	Feasibility Problem
d2	S4	6,7	Efficiency
d2	S5	8	Complexity
d2	S5	9	Necessary conditions
d2	S6	10	Envelope theorem
d3	S1	1,2,7	Global warming
d3	S2	3,4,5	Observed changes
d3	S3	6,7	Green house gases
d3	S4	8	Solar activity
d3	S5	9,10,11	Feedback
d3	S6	12,13,14,15	Climate models
d3	S7	16,17	Expected environmental effects
d3	S8	18	Ecological systems

Table 4 shows the identified key words for this real time corpus and the DOCIDs of Documents containing the keywords. All these pages were cleaned by removing unrelevant HTML tags(like meta tag not contributing to content mining) and segmentation algorithm was applied. Segment details were saved in relational database. Document di contains n number of segments as di = {s1,s2,…sn}. Only m segments are selected for summarization. This improves the processing efficiency of the information server by (n-m)/n which is a remarkable improvement in view of the processing load to the server at query time.

Our data collection process was carried out using Google Search Engine. Of the five major or "core" search engines, Google held a substantial lead over its rivals for more than the past five years (Pasupathi *et al*., 2011) (according to comScore research and Search Marketing Standard). Ebiz MBA Knowledge database statistics says that more than 9 billion monthly visitors are using Google for information search on WWW. For testing purpose we took 15 web documents from the real time corpus on which the segmentation algorithm was applied. Segment details (for only 3 documents) were stored in the Segment Table as shown in **Table 5**.

3. RESULTS

In the above mentioned example document d1 contains 3 segments and 8 nodes. If the query string given is "engineering college + traning and Placement" then during summarization only segments s2 and s4 of document d1 need to be processed for summarization. This improves the processing efficiency by 2/4 at the segment level and 6/9 at the node level. Starting from these set of leaf nodes the specific branch of the DOM tree can be considered for generating the summary. The efficiency improves for larger web documents as the segment required to be processed will be remarkably less.

4. DISCUSSION

The **Table 6** shows that the segmentation helps to improve the summarization efficiency considerably which in also depicted in the following graph. Instead of processing the complete document only a part that is few segments relevant to the query alone are going to be processed.

Figure 3 clearly indicates that the pre-processing segmentation improves the summarization efficiency by reducing the size of text units to be processed for generating summary.

Table 6. Comparison between summarization using and without using Segments

Query string	DOCID identified	DOC selected for summarization	No. of segments in the document	No. of nodes in the document	No. of relevant segments	No. of relevant nodes	Segment level improvement %	Node level improvement %	Nodes to be processed without segments
Engineering College + placement & training	d1,d4,d5,d7	d1	3	7	2	6	33.33	33.33	9
Multi objective optimization	d2,d7,d8	d2	6	10	1	2	83.33	80.00	10
Complexity of optimization	d2,d8,d9	d2	6	10	2	1	66.67	90.00	10
Green house gases	d3,d12	d3	8	17	1	2	87.50	88.24	17
Climate Models	d3,d14,d15	d3	8	17	1	4	87.50	76.47	17

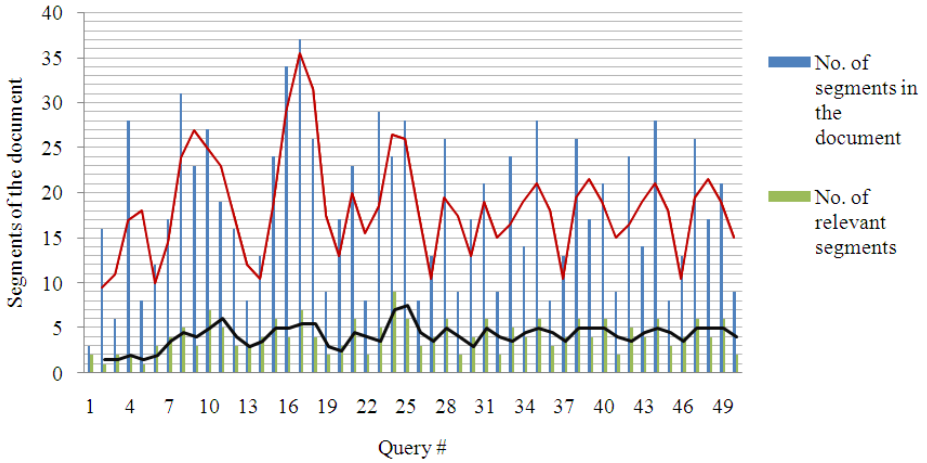

Fig. 3. Increase in Processing Efficiency of Summarization System when using Segments (No. of queries Qn = 50)

As the number of nodes in the document increases the efficiency of summarization using segmentation increases and the time complexity and processing overhead of the server is drastically decreased. Summarization without using segments needs to process all nodes of the document which in turn will increase time complexity of the process.

Segmentation as pre-processing for summarization is an innovative idea which has not yet been applied in any summarization system.

5. CONCLUSION

Query based summarization focuses on extracting query relevant pieces of information from the web page at query time. Information servers need to process the entire content of selected web pages to compose the summary page. This study proposed an innovative idea of identifying relevant sentence from the web page as segments and materializing the segment information in relational database during pre-processing stage i.e., offline. Web documents were segmented based on frequent term sets and WordNet distance between term sets. Query relevant segments of the user selected URL from the search result were identified and considered for summary generation process. This reduces the load for information servers to produce on the fly summaries at query time. Query relevant summary is really a boon to information seekers who need to understand the content of the web page quickly.

Pre-processed segments are more helpful in reducing processing overload of the information servers by reducing the scope of summarization to few relevant segments instead of processing the entire document at query time. In this scenario, the size of the document does not have much impact on the summarization process.

6. REFERENCES

1. Cai, D., S. Yu, J.R. Wen and W.Y. Ma, 2003. VIPS: A vision-based page segmentation algorithm. Microsoft Corporation.
2. Chen, Z. and J. Shen, 2009. Research on query-based automatic summarization of webpage. ISECS Int. Colloquium Comput. Commun.

Control Manage., 1: 173-176. DOI: 10.1109/CCCM.2009.5270475
3. Chien, J.T. and C.H. Chueh, 2012. Topic-based hierarchical segmentation. IEEE Trans. Audio, Speech Language Process., 20: 55-66. DOI: 10.1109/TASL.2011.2143405
4. Chitra, P., R. Baskaran and K. Sarukesi, 2011. Query sensitive comparative summarization of search results using concept based segmentation. Comput. Sci. Eng. Int. J., 1: 31-43. DOI: 10.5121/cseij.2011.1503
5. Feng, J., G. Li and J. Wang, 2011. Finding top-k answers in keyword search over relational databases using tuple units. IEEE Trans. Knowl. Data Eng., 23: 1781-1794. DOI: 10.1109/TKDE.2011.61
6. Hao, D., W. Zuo, T. Peng and F. He, 2011. An approach for calculating semantic similarity between words using WordNet. Proceedings of the 2nd International Conference on Digital Manufacturing and Automation, Aug. 5-7, IEEE Xplore Press, Zhangjiajie, Hunan, pp: 177-180. DOI: 10.1109/ICDMA.2011.50
7. Iosif, E. and A. Potamianos, 2010. Unsupervised semantic similarity computation between terms using web documents. IEEE Trans. Knowl. Data Eng., 22: 1637-1647. DOI: 10.1109/TKDE.2009.193
8. Islam, A. and D. Inkpen, 2008. Semantic text similarity using corpus-based word similarity and string similarity. ACM Trans. Knowl. Discovery Data, 2: 1-25. DOI: 10.1145/1376815.1376819
9. Kohlschutter, C. and W. Nejdl, 2008. A densitometric approach to web page segmentation. Proceedings of the ACM 17th Conference on Information and Knowledge Management, Oct. 26-30, ACM Press, Napa Valley, CA, USA., pp: 1173-1182. DOI: 10.1145/1458082.1458237
10. Kogilavani, 2012. Sentence annotation based enhanced semantic summary generation from multiple documents. Am. J. Applied Sci., 9: 1063-1070. DOI: 10.3844/ajassp.2012.1063.1070
11. Kumar, Y.J. 2011. Automatic Multi Document Summarization Approaches. J. Comput. Sci., 8: 133-140. DOI: DOI: 10.3844/jcssp.2012.133.140
12. Kuppusamy, K.S. and G. Aghila, 2012. A personalized web page content filtering model based on segmentation. Int. J. Inform. Sci. Technol., 2: 41-51. DOI: 10.5121/ijist.2012.2104
13. Landauer, T.K. and S.T. Dumais, 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. Psychol. Rev., 104: 211-240. DOI: 10.1037/0033-295X.104.2.211
14. Landauer, T.K., P.W. Foltz and D. Laham, 1998. An introduction to latent semantic analysis. Dis. Proce., 25: 259-284. DOI: 10.1080/01638539809545028
15. Li, Y., D. Mclean, Z.A. Bandar, J.D. O'Shea and K. Crockett, 2006. Sentence similarity based on semantic nets and corpus statistics. IEEE Trans. Knowl. Data Eng., 18: 1138-1150. DOI: 10.1109/TKDE.2006.130
16. Mohamed, A.A. and S. Rajasekaran, 2006. Improving query-based summarization using document graphs. Proceedings of the IEEE International Symposium on Signal Processing and Information Technology, (SPIT' 06), IEEE Xplore Press, Vancouver, BC., pp: 408-410. DOI: 10.1109/ISSPIT.2006.270835
17. Pasupathi, C., B. Ramachandran and S. Karunakaran, 2011. Selection based comparative summarization of search results using concept based segmentation. Trends Netw. Commun., 97: 655-664. DOI: 10.1007/978-3-642-22543-7_67
18. Pnueli, A., R. Bergman, S. Schein and O. Barkol, 2009. Web page layout via visual segmentation. Hewlett-Packard Development Company, L.P. http://www.hpl.hp.com/techreports/2009/HPL-2009-160.pdf
19. Shehata, S., F. Karray and M.S. Kamel, 2010. An efficient concept-based mining model for enhancing text clustering. IEEE Trans. Knowl. Data Eng., 22: 1360-1371. DOI: 10.1109/TKDE.2009.174
20. Wang, F.L., T.L. Wong, A.N.H. Mak, 2008. Organization of documents for multiple document summarization. Proceedings of the 7th IEEE International Conference on Web-Based Learning, Aug. 20-22, IEEE Xplore Press, Jinhua, pp: 98-104. DOI: 10.1109/ICWL.2008.6

21. Yen, C.C. and J.S. Hsu, 2009. Associated Pagerank: Improved pagerank measured by frequent term sets. Proceedings of the IEEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurements Systems, May 11-13, IEEE Xplore Press, Hong Kong, pp: 282-286. DOI: 10.1109/VECIMS.2009.5068909