Summarizing Web Sites Automatically

Y. Zhang, N. Zincir-Heywood, and E. Milios

Dalhousie University

Abstract. This research is directed towards automating the Web Site summarization task. To achieve this objective, an approach, which applies machine learning and natural language processing techniques, is employed. The automatically generated summaries are compared to manually constructed summaries from DMOZ Open Directory Project. The comparison is performed via a formal evaluation process involving human subjects. Statistical evaluation of the results demonstrates that the automatically generated summaries are as informative as human authored DMOZ summaries and significantly more informative than home page browsing or time limited site browsing.

1 Introduction

The information overload problem [17] on the World Wide Web has brought users great difficulty to find useful information quickly and effectively. It has been more and more difficult for the user to skim over a Web site and get an idea of its contents. Currently, manually constructed summaries by volunteer experts are available, such as the DMOZ Open Directory Project [1]. These human-authored summaries give a concise and effective description of popular Web sites. However, they are subjective, and expensive to build and maintain [8]. Hence in this work, our objective is to summarize the Web site automatically.

The technology of automatic summarization of text is maturing and may provide a solution to this problem [17, 16]. Automatic text summarization produces a concise summary by abstraction or extraction of important text using statistical approaches [9], linguistic approaches [4] or combination of the two [5, 13, 16].

The goal of abstraction is to produce coherent summaries that are as good as human authored summaries [13]. To achieve this, extraction systems analyze a source document to determine significant sentences, and produce a concise summary from these significant sentences [19].

Basically Web page summarization derives from text summarization techniques [9]. However, it is a great challenge to summarize Web pages automatically and effectively [3], because Web pages differ from traditional text documents in both structure and content. Instead of coherent text with a well-defined discourse structure, Web pages often have diverse contents such as bullets and images [6].

Currently there is no effective way to produce unbiased, coherent and informative summaries of Web pages automatically. Amitay et al [3] propose a unique approach, which relies on the hypertext structure. This approach is applied to "generate short coherent textual snippets presented to the user with search engine results".

Garcia-Molina et al [9] compare alternative methods to summarize Web pages for display on handheld devices. They test the performance of these methods by asking human subjects to perform specific tasks using each method, and conclude that the combined *Keyword/Summary* method provides the best performance in terms of access times and number of pen actions on the hand held devices.

Our objective is to automate summarization of Web sites, not simply Web pages. To this end, the "Keyword/Summary" idea of [9] is adopted. However, this methodology is enhanced by applying machine learning and natural language processing techniques. A summary is produced in a sequence of stages: URL & Text extraction are described in Section 2. Sections 3, 4 and 5 detail the narrative paragraph, key-phrase and key-sentence extraction, respectively. Evaluation results are given in Section 6 and conclusions are drawn in Section 7.

2 URL and Text Extraction

Since our objective is to summarize the Web site, we want to focus on top-level pages in order to extract the contents which describe the Web site in a general sense. A module called *Site Crawler* was developed, that crawls within a given Web site using *breadth-first-search*. This means that only Web pages physically located in this site will be crawled and analyzed. Besides tracking the URLs of these Web pages, the Site Crawler also records the depth (i.e. level) and length of each page. Depth represents the number of "hops" from the home page to the current page. For example, if we give the home page depth 1, then all pages which can be reached by an out-link of the home page are assigned depth 2. Length of a Web page is the number of characters in the Web page source file. The Site Crawler only keeps known types of Web pages, such as .htm, .html, .shtml, .php, etc. Handling other types of text and non-text files is a topic for future research.

Normally the Site Crawler crawls the top 1000 pages of a Web site, according to a breadth-first traversal starting from the home page. The number of pages to crawl (1000) is based on the observation after crawling 60 Web sites (identified in DMOZ subdirectories), that there is an average of 1000 pages up to and including depth equal to 4. For each Web site, the Site Crawler will stop crawling when either 1000 pages have been collected, or it has finished crawling depth 4, whichever comes first.

After the URLs of the top 1000 Web pages are collected, the plain text must be extracted from these pages. In this work the text browser *Lynx* [10] is used for this purpose.

3 Narrative Paragraph Classification

The summary of the Web site will be created on the basis of the text extracted by Lynx. However, Web pages often do not contain a coherent narrative structure [6], so our aim is to identify rules for determining which text should be considered for summarization and which should be discarded. This is achieved in two steps: First, criteria are defined for determining if a paragraph is long enough to be considered for analysis. Then, additional criteria are defined to classify long paragraphs into narrative or non-narrative. Only narrative paragraphs are used in summary generation. The criteria are defined automatically using supervised machine learning.

Intuitively, whether a paragraph is long or short is determined by its length (i.e., the number of characters). However, two more features, number of words, and number of characters in all words, might also play a key role. In order to determine which feature is the most important, a total of 700 text paragraphs is extracted from 100 Web pages. Statistics of three attributes *Length, NumberOfWords* and *NumberOfChars* are recorded from each paragraph. *Length* is the number of all characters in the paragraph. *NumberOfWords* is the number of words in this paragraph, and *NumberOfChars* is the total number of characters

in all words. Then each text paragraph is labelled as *long* or *short* manually. The decision tree learning program C5.0 [2] is used to construct a classifier, *LONGSHORT*, for this task.

The training set consists of 700 instances. Among the 700 cases, there are 36 cases misclassified, leading to an error of 5.1%. The cross-validation of the classifier is listed in Table 1. The mean error rate 5.9% indicates the classification accuracy of this classifier.

Table 1. Cross-validation of C5.0 classifier *LONGSHORT*

Fold	1	2	3	4	5	6	7	8	9	10	Mean
Size	2	2	2	2	2	2	2	2	2	2	2.0
Error(%)	5.7	5.7	11.4	4.3	2.9	4.3	4.3	7.1	2.9	10.0	5.9

Not all long paragraphs provide coherent information in terms of generating a meaningful summary. Informally, whether a paragraph is *narrative* or *non-narrative* is determined by the coherence of its text. Our hypothesis is that the frequencies of the part-of-speech tags of the words in the paragraph contain sufficient information to classify a paragraph as narrative. To test this hypothesis, a training set is generated as follows: First, 1000 Web pages were collected from DMOZ subdirectories, containing a total of 9763 text paragraphs, among which a total of 3243 paragraphs were classified as long. Then, the part-of-speech tags for all words in these paragraphs are computed using a rule-based part-of-speech tagger [7].

After part-of-speech tagging, the following attributes are extracted from each paragraph. Let n_i (i = 1, 2, ... , 32) be the number of occurrences of tag i, and S be the total number of tags (i.e. words) in the paragraph. Let P_i be the fraction of S, that n_i represents.

$$\begin{aligned} S &= \sum_{i=1}^{32} n_i \\ P_i &= n_i/S \ \ (i = 1, 2, ..., 32) \ . \end{aligned} \tag{1}$$

A total of 34 attributes are associated with each paragraph in the training set. The length of the paragraph in characters, and the length of the paragraph in words are added to the 32 attributes P_1, P_2, ..., P_{32}, as defined in (1). Then each paragraph is manually labelled as *narrative* or *non-narrative*. Finally, a C5.0 classifier *NARRATIVE* is trained on the training set of 3243 paragraphs.

Among the 3242 cases, about 63.5% of them are following this rule: if the percentage of *Symbols* is less than 6.8%, and the percentage of *Preposition* is more than 5.2%, and the percentage of *Proper Singular Nouns* is less than 23.3%, then this paragraph is *narrative*. There are 260 cases misclassified, leading to an error

of 8.0%. The cross-validation of the classifier *NARRATIVE* is listed in Table 2. The mean error rate 11.3% indicates the predictive accuracy of this classifier.

Table 2. Cross-validation of C5.0 classifier *NARRATIVE*

Fold	1	2	3	4	5	6	7	8	9	10	Mean
Size	5	5	3	4	4	5	4	3	4	3	4.0
Error	11.1	9.3	13.6	11.1	9.9	7.4	9.3	16	10.5	14.7	11.3

4 Key-Phrase Extraction

Traditionally, key-phrases (key-words and key-terms) are extracted from the document in order to generate a summary. Key-phrase extraction from a body of text relies on an evaluation of the importance of each phrase [9]. In terms of automatically summarizing a Web site, a phrase is considered as *key-phrase*, if and only if it occurs very frequently in the Web pages of the site, i.e., the total frequency is very high.

In this work, a *key-phrase* can be either *key-word* or *key-term*. *Key-word* is a single word with very high frequency over the set of Web pages, and *key-term* is a two-word term with very high frequency.

As we discussed in the previous section, Web pages are quite different from traditional documents. The existence of *anchor text* and *special text* contributes much to the difference. *Anchor text* is the text of hyper links, and it "often provides more accurate descriptions of Web pages than the pages themselves" [8]. *Special text* includes title, headings and bold or italicized text. The assumption is that both anchor text and special text may play a key role in describing important topics of Web pages. Therefore a supervised learning approach is applied to test this assumption.

In order to determine the key-words of a Web site, a decision tree is produced. A data set of 5454 candidate key-words (at most 100 for each site) from 60 Web sites are collected. The sites are taken from DMOZ subdirectories. For each site, the frequencies of each word in narrative text, anchor text and special text (denoted by fn, fa and fs, respectively), are measured. Then the total frequency, f, of each word over these three categories is computed, where the weight for each category is the same. Moreover, it should be noted that 425 stop words (*a, about, above, across, after, again, against, all, almost, alone, along, already, also, although, always, among, an, and, ...*) [11] are discarded in this stage. Then a simple stemming process was applied to identify each singular noun and its plural form. For example, *product* : 2100 and *products* : 460 yields *product* : 2560.

After this process, on the average there were about 5,100 different words (excluding stop words) within the text body of the top 1000 Web pages. Figure

1 shows that the rank and frequency statistics of these words fit Zipf's Law [15]. The words with the lowest frequencies are obviously not key-words, hence only those words whose frequency is more than 5% of the maximum frequency are kept as *candidate key-words*. This step eliminates about 98% of the original words, leaving about 102 candidate key-words per site. As a result, the top 100 candidate key-words are kept and nine features of each candidate key-word C_i are defined, as shown in Table 3. The feature *Tag* was obtained by tagging candidate key-words with rule-based part-of-speech tagger [7].

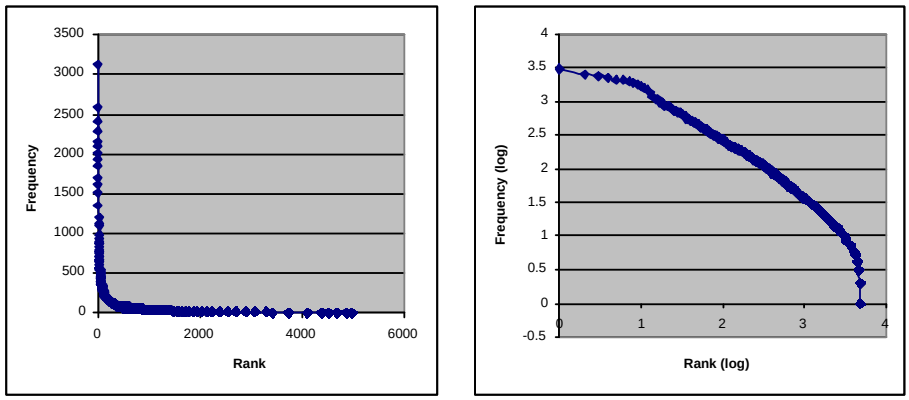

Fig. 1. Rank-Frequency data and Zipf's Law

Table 3. Feature list of candidate key-words

No.	Feature	Value	Meaning
1	W	$W_i{=}f_i/\sum_{i=1}^{100} f_i$	Weight of candidate key-word
2	R	$R_i{=}f_i/\max_{i=1}^{100} f_i$	Ratio of frequency to max freq.
3	WN	$WN_i{=}fn_i/\sum_{i=1}^{100} fn_i$	Weight in *narrative* text only
4	RN	$RN_i{=}fn_i/\max_{i=1}^{100} fn_i$	Ratio in *narrative* text only
5	WA	$WA_i{=}fa_i/\sum_{i=1}^{100} fa_i$	Weight in *anchor* text only
6	RA	$RA_i{=}fa_i/\max_{i=1}^{100} fa_i$	Ratio in *anchor* text only
7	WS	$WS_i = fs_i/\sum_{i=1}^{100} fs_i$	Weight in *special* text only
8	RS	$RS_i = fs_i/\max_{i=1}^{100} fs_i$	Ratio in *special* text only
9	Tag	$CC, CD, ..., WRB$	Part-of-speech tag ([7])

Next, each candidate key-word is labelled manually as *key-word* or *non-key-word*. The criterion to determine if a candidate key-word is a true key-word is that a key-word provides important information which is related with the Web

site. Based on frequency statistics and part-of-speech feature of these candidate key-words, a C5.0 classifier *KEY-WORD* is constructed.

Among the total 5454 cases, 222 cases are misclassified, leading to an error of 4.1%. In the decision tree, about 35% of cases are following this rule: if R (defined as the ratio of a candidate key-word's frequency to the maximum frequency in Table 3) is less than or equal to 0.1, then this candidate key-word is a non-key-word. Another main stream of cases follows the second rule: if R is greater than 0.1, and part-of-speech tag is NN (common singular nouns [7]), and RA (ratio in anchor text) is less than or equal to 0.798, then the candidate key-word is a key-word. This case covers 45% of the data set.

The most important rule here is: if R is greater than 0.1 and part-of-speech tag is NN (common singular nouns) or VBG (verb -ing [7]), then WA (weight in anchor text), RA (ratio in anchor text) and/or WS (weight in special text) will determine if a candidate key-word should be classified as key-word or non-key-word. This demonstrates that our assumption is true, i.e., anchor text and special text do play important roles in determining key-words of a Web site. The cross-validation results of the classifier *KEY-WORD* is listed in Table 4. The mean error rate 4.9% indicates the predictive accuracy of this classifier.

Table 4. Cross-validation of C5.0 classifier *KEY-WORD*

Fold	1	2	3	4	5	6	7	8	9	10	Mean
Size	22	20	20	30	23	18	20	27	20	20	22.0
Error(%)	4.0	5.1	5.5	4.4	4.0	5.1	5.1	5.9	5.5	4.0	4.9

Furthermore, it is observed that terms which consist of two of the top 100 candidate key-words may exist with high frequency. Such a term could be good as part of the description of the Web site. Thus, a similar approach with *automatic key-word extraction* is developed to identify key-terms of the Web site.

The algorithm combines any two of the top 100 candidate key-words and searches for these terms in collocation over narrative text, anchor text and special text. Then these terms are sorted by frequency and the top 30 are kept as *candidate key-terms*. A C5.0 classifier *KEY-TERM* is constructed based on frequency statistics and tag features of 1360 candidate key-terms, which were extracted from 60 Web sites (collected from DMOZ subdirectories). The C5.0 classifier *KEY-TERM* is similar to the KEY-WORD classifier except that it has two part-of-speech tags *Tag1* and *Tag2*, one for each component word.

Once the decision tree rules for determining key-terms have been built, they are applied for automatic key-term extraction to the Web pages of a Web site. The top 10 key-terms (ranked by total frequency) for each site are kept as part of the summary. The frequency of candidate key-words is reduced by subtracting the frequency of top 10 key-terms, which includes them. Then the KEY-WORD classifier is applied. Finally, the top 25 key-words (ranked by frequency) are kept

as part of the summary. It is observed that 40% to 70% of key-words and 20% to 50% of key-terms appear in the home page of a Web site.

5 Significant Sentence Extraction

Once the key-words and key-terms are identified, the most significant sentences can be retrieved from all narrative paragraphs. Each sentence is assigned a significance factor or sentence weight. The top five sentences, ranked according to sentence weight, are chosen as part of the summary. In order to achieve this goal, a modified version of the procedure in [9] is applied.

First, the sentences containing any of the list L of key-phrases, consisting of the top 25 key-words and top 10 key-terms identified previously, are selected. Second, all clusters in each selected sentence S are identified. A *cluster* C is a sequence of consecutive words in the sentence for which the following is true: (1) the sequence starts and ends with a key-phrase in L, and (2) less than D non-key-phrases must separate any two neighboring key-phrases within the sentence. D is called the "distance cutoff", and we used a value of 2 as in [9]. Third, the weight of each cluster within S is computed. The maximum of these weights is taken as the sentence weight. As shown in Table 5, a cluster's weight is computed by adding the weights of all key-phrases within the cluster, and dividing this sum by the total number of key-phrases within the cluster. The weight of key-phrase i is defined as $W_i = f_i / \sum_{i=1}^{100} f_i$, where f_i is the frequency of the key-phrase in the Web site (Table 3).

Table 5. Example of clustering

Candidate Sentence			
The Software Engineering Information Repository (SEIR) is a Web-based repository of information on software engineering practices that lead to improved organizational performance.			
Key-Phrase	**Weight**	**Cluster**	**Weight**
information	0.021	1. Software Engineering Information	0.157
software engineering	0.293	2. information on software engineering practices	0.109
practice	0.013	Sentence Weight: 0.157	

The weights of all sentences in *narrative* text paragraphs are computed and the top five sentences ranked according to sentence weights are included in the summary as *key-sentences*. Finally, a summary is formed consisting of the top 25 key-words, top 10 key-terms and top 5 key-sentences.

Table 6 shows a summary example generated by our system for the Software Engineering Institute (SEI) Web site. This summary gives a brief description of SEI's mission and various activities, whereas Table 7 shows the DMOZ summary for the same Web site.

Table 6. Automatically created summary of Software Engineering Institute Web site

Part 1. Top 25 Key-words				
sei	system	software	cmu	product
component	information	process	architecture	organization
course	program	report	practice	project
method	design	institute	development	research
document	management	defense	technology	team

Part 2. Top 10 Key-terms				
software engineering	carnegie mellon	development center	software process	software architecture
maturity model	risk management	software development	process improvement	software system

Part 3. Top 5 Key-sentences
1. The Software Engineering Information Repository (SEIR) is a Web-based repository of information on software engineering practices that lead to improved organizational performance.
2. Because of its mission to improve the state of the practice of software engineering, the SEI encourages and otherwise facilitates collaboration activities between members of the software engineering community.
3. The SEI mission is to provide leadership in advancing the state of the practice of software engineering to improve the quality of systems that depend on software.
4. The Software Engineering Institute is operated by Carnegie Mellon University for the Department of Defense.
5. The Software Engineering Institute (SEI) sponsors, co-sponsors, and is otherwise involved in many events throughout the year.

As we can see, the automatically generated summary basically covers the key contents described by human authors.

6 Experiments and Evaluation

In order to measure the overall performance of our approach, four sets of experiments were performed. During these experiments, automatically generated summaries are compared with human-authored summaries, home page browsing and time-limited site browsing, to measure their performance in a specific task.

From the DMOZ Open Directory Project, 20 manually constructed summaries were selected from four subdirectories. As listed in Table 8, sites 1-5 are in the *Software/Software Engineering*[1] subdirectory. Sites 6-10 are in the *Artificial Intelligence/Academic Departments*[2] subdirectory. Sites 11-15 are in *Major Companies/Publicly Traded*[3] subdirectory. And finally sites 16-20 are in

[1] http://dmoz.org/Computers/Software/Software_Engineering/
[2] http://dmoz.org/Computers/Artificial_Intelligence/Academic_Departments/
[3] http://dmoz.org/Business/Major_Companies/Publicly_Traded/

Table 7. DMOZ summary of Software Engineering Institute Web site

Software Engineering Institute (SEI) - SEI is a federal research center whose mission is to advance the state of the practice of software engineering to improve the quality of systems that depend on software. SEI accomplishes this mission by promoting the evolution of software engineering from an ad hoc, labor-intensive activity to a discipline that is well managed and supported by technology.

E-Commerce/Technology Vendors[4] subdirectory. These sites were selected randomly and are of varying size and focus.

Our approach, *W3SS* (World Wide Web Site Summarization), is used to create summaries of these 20 Web sites. Each W3SS summary consists of the top 25 key-words, the top 10 key-terms and the top 5 key-sentences.

Table 8. URL list of the Web sites used in the experiments

Subdirectory	**Site URL**
Software/ Software Engineering	1. http://case.ispras.ru 2. http://www.ifpug.org 3. http://www.mapfree.com/sbf 4. http://www.cs.queensu.ca/Software-Engineering 5. http://www.sei.cmu.edu
Artificial Intelligence/ Academic Departments	6. http://www.cs.ualberta.ca/~ai 7. http://www.ai.mit.edu 8. http://www.aiai.ed.ac.uk 9. http://www.ai.uga.edu 10. http://ai.uwaterloo.ca
Major Companies/ Publicly Traded	11. http://www.aircanada.ca 12. http://www.cisco.com 13. http://www.microsoft.com 14. http://www.nortelnetworks.com 15. http://www.oracle.com
E-Commerce/ Technology Vendors	16. http://www.adhesiontech.com 17. http://www.asti-solutions.com 18. http://www.commerceone.com 19. http://www.getgamma.com 20. http://www.rdmcorp.com

There are two major types of summarization evaluations: *intrinsic* and *extrinsic* [14, 17]. Intrinsic evaluation compares automatically generated summaries against a gold standard (ideal summaries). Extrinsic evaluation measures the performance of automatically generated summaries in a particular task (e.g.,

[4] http://dmoz.org/Business/E-Commerce/Technology_Vendors/

classification). Extrinsic evaluation is also called task-based evaluation and it has become more and more popular recently [18]. In this work, extrinsic evaluation is used.

In extrinsic evaluation, the objective is to measure how informative W3SS summaries, DMOZ summaries, home page browsing and time-limited site browsing are in answering a set of questions [21] about the content of the Web site. Each question is meant to have a well-defined answer, ideally explicitly stated in the summary, rather than being open-ended. Four groups of graduate students in Computer Science (5 in each group) with strong World Wide Web experience were asked to take the test as follows:

The first and second group was asked to read each W3SS and DMOZ summary, respectively and then answer the questions. The third group was asked to browse the home page of each of the 20 Web sites and answer the questions. The last group was asked to browse each Web site for at most 10 minutes (time-limited site browsing) and answer all questions. All answers were then graded in terms of their quality in a scale 0-20. The grades are tabulated in [21].

The average score of the five subjects working with the W3SS summaries is 15.0 out of a possible 20. Moreover, the variance between the average scores of all summaries over five subjects is only 0.213, which shows that all subjects in this experiment evaluated W3SS summaries consistently.

The average score of the five subjects working with the DMOZ summaries is 15.3 out of 20, hence the overall performance of DMOZ summaries is slightly better than that of W3SS ones (with an overall average 15.0). The variance between the average scores of all DMOZ summaries over five subjects is 1.267, much larger than that of W3SS summaries. As indicated in Fig. 2, there are 11 Web sites whose W3SS summaries are better than DMOZ summaries, and 8 sites whose W3SS summaries are worse than DMOZ summaries. The remaining site has the same quality of W3SS and DMOZ summary.

In the *home page browsing* experiment, every subject was allowed to browse only the home page, and there are a few very poor marks as low as 4.4 and 5.0. The average score of the five subjects browsing home pages is 12.7 out of 20, which is less than 15.0 of W3SS summaries and 15.3 of DMOZ summaries. As indicated in Fig. 3, the home page alone is often not sufficiently informative, and that digging deeper into the site conveys more complete information about the site than the home page alone. In order to understand the site better, more browsing beyond the home page alone is needed.

In the fourth test, each subject was allowed 10 minutes to browse each Web site, and look for the answers of all questions. For each site, the average score of all subjects varies from 7.0 to 20.0. This implies that either some Web sites were poorly designed, or there is too much non-text (e.g., flash) in top-level pages, which may confuse the user's understanding of the site. The average score of the five subjects browsing the sites is 13.4 out of 20, which is less than that of both W3SS and DMOZ summaries. As indicated in Fig. 4, it is not so easy to get a good understanding of the site's main contents by browsing within a

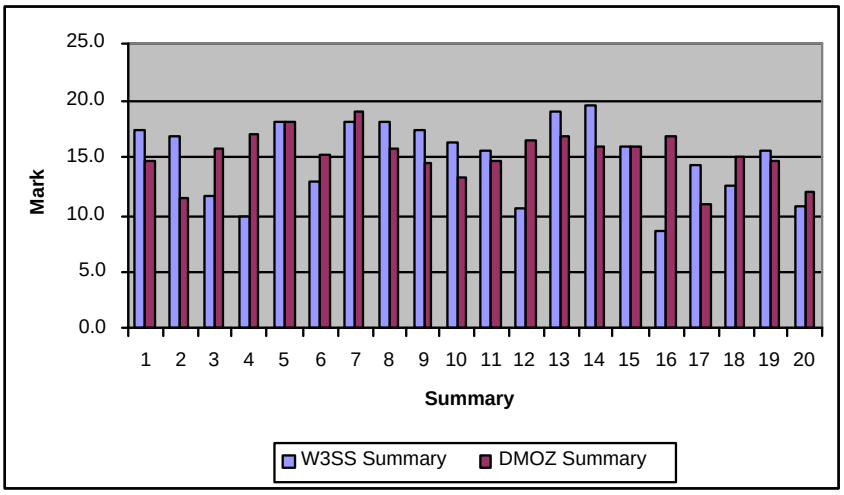

Fig. 2. W3SS summaries vs. DMOZ summaries

limited time period. This indicates that our approach of automatically creating summaries is potentially useful because it saves the reader much time.

To confirm the above intuitive conclusions, we perform a two-factor Analysis of Variance with replications on the raw scores from the above experiments. As shown in Table 9, there is no significant difference between our summaries and the human-authored summaries, and between home-page and time-limited site browsing. However, our summaries and the human-authored summaries are significantly better than home-page and time-limited site browsing.

Table 9. Pairwise ANOVA results for the four experiments. W3SS, DMOZ, HPB, TLSP is the performance of our summaries, the human-authored summaries, home-page browsing and time-limited site browsing.

	W3SS	DMOZ	HPB
DMOZ	$F_{1,190} = 0.18$ $Pvalue = 0.67$		
HPB	$F_{1,190} = 17.42$ $Pvalue < 0.0001$	$F_{1,190} = 23.7$ $Pvalue < 0.0001$	
TLSB	$F_{1,190} = 6.13$ $Pvalue = 0.014$	$F_{1,190} = 8.88$ $Pvalue = 0.003$	$F_{1,190} = 1.62$ $Pvalue = 0.20$

Since the W3SS summaries are as informative as DMOZ summaries, they could be transformed into proper prose by human editors without browsing the

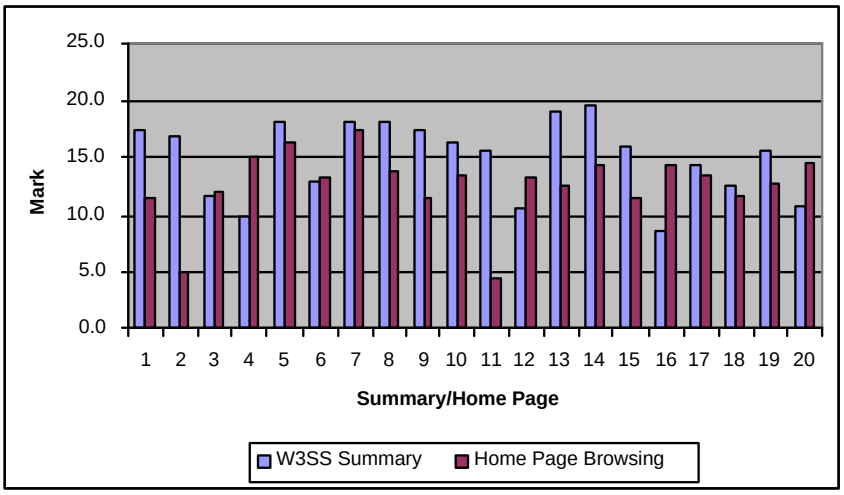

Fig. 3. W3SS summaries vs. Home page browsing

Web site. Automating such a transformation is beyond the state of the art of natural language processing.

7 Conclusion and Discussion

In this work, we developed a new approach for generating summaries of Web sites. Our approach relies on a Web crawler that visits Web sites and summarizes them off-line. It applies machine learning and natural language processing techniques to extract and classify narrative paragraphs from the Web site, from which key-phrases are then extracted. Key-phrases are in turn used to extract key-sentences from the narrative paragraphs that form the summary, together with the top key-phrases. We demonstrate that our summaries, although not in proper prose, are as informative as human-authored summaries, and significantly better than browsing the home page or the site for a limited time. Our approach should be easy to transform into proper prose by human editors without having to browse the Web site. The performance of our method depends on the availability of sufficient narrative content in the Web site, and the availability of explicit narrative statements describing the site.

However, several issues need to be addressed to further improve the performance of our approach. Currently the top 1000 (or all pages between depth 1 and depth 4, inclusively) Web pages of a Web site are crawled for text extraction. Supervised learning may be used instead to determine the most appropriate number of pages to crawl.

In the key-term extraction step, we simply combine any two of top 100 candidate key-words. More sophisticated methods, such as the C-value/NC-value

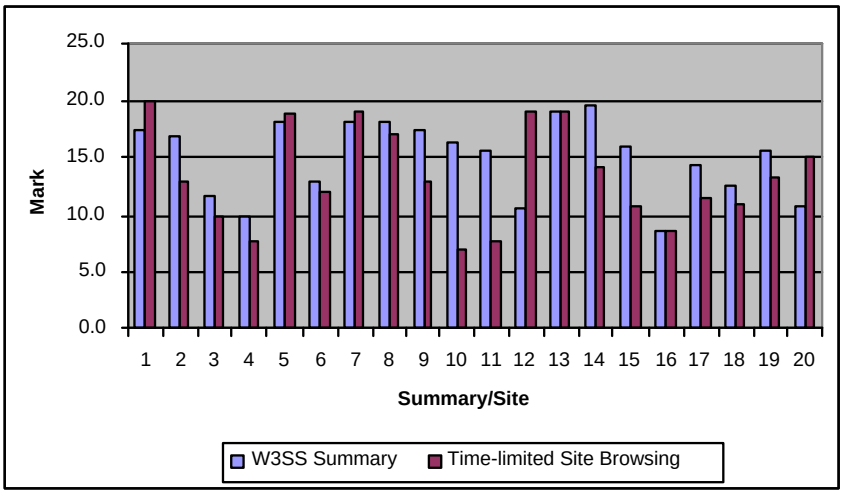

Fig. 4. W3SS summaries vs. Time-limited site browsing

method [12] will be considered to automatically recognize multi-word terms. Also further research is required to determine appropriate weights for the key-phrases from different categories (plain text, anchor text and special text). And redesign of the evaluation process to reduce the inter-rater reliability problem [20] is a topic for future research. Intrinsic evaluation should also be considered.

Acknowledgements. We are thankful to Prof. Michael Shepherd for many valuable suggestions on this work, and to Jinghu Liu for suggesting the use of Lynx for text extraction from Web pages. The research has been supported by grants from the Natural Sciences and Engineering Research Council of Canada.

References

[1] Netscape 1998-2002. *DMOZ - Open Directory Project.* http://dmoz.org, last accessed on Oct. 9, 2002.

[2] RULEQUEST RESEARCH 2002. *C5.0: An Informal Tutorial.* www.rulequest.com/see5-unix.html, last accessed on Oct. 9, 2002.

[3] E. Amitay and C. Paris. Automatically summarising web sites - is there a way around it? In *ACM 9th International Conference on Information and Knowledge Management*, 2000.

[4] C. Aone, M.E. Okurowski, J. Gorlinsky, and B. Larsen. A scalable summarization system using robust NLP. In *Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization*, pages 66–73, 1997.

[5] R. Barzilay and M. Elhadad. Using lexical chains for text summarization. In *Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97), ACL, Madrid, Spain*, 1997.

[6] A. Berger and V. Mittal. Ocelot: a system for summarizing web pages. In *Proceedings of SIGIR*, pages 144–151, 2000.

[7] E. Brill. A simple rule-based part of speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing, ACL*, 1992.

[8] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *7th International World Wide Web Conference*, 1998.

[9] O. Buyukkokten, H. Garcia-Molina, and A. Paepcke. Seeing the whole in parts: Text summarization for web browsing on handheld devices. In *Proceedings of 10th International World-Wide Web Conference*, 2001.

[10] Internet Software Consortium. *Lynx: a World Wide Web (WWW) client for cursor-addressable, character-cell display devices.* lynx.isc.org, last accessed on Oct. 9, 2002.

[11] C. Fox. *Lexical analysis and stoplists, In W. Frakes and R. Baeza-Yates, editors, Information Retrieval: Data Structures & Algorithms.* Prentice Hall, Englewood Cliffs, NJ, chapter 7, pages 102?30, 1992.

[12] K. Frantzi, S. Ananiadou, and H. Mima. Automatic recognition of multiword terms. *International Journal of Digital Libraries*, 3(2):117–132, 2000.

[13] J. Goldstein, M. Kantrowitz, V. Mittal, and J. Carbonell. Summarizing text documents: Sentence selection and evaluation metrics. In *Proceedings of SIGIR*, pages 121–128, 1999.

[14] S. Jones and J. Galliers. *Evaluating Natural Language Processing Systems: an Analysis and Review.* Springer, New York, 1996.

[15] Wentian Li. *Zipf's Law.* linkage.rockefeller.edu/wli/zipf, last accessed on Oct. 9, 2002.

[16] I. Mani. Recent developments in text summarization. In *ACM Conference on Information and Knowledge Management, CIKM'01*, pages 529–531, 2001.

[17] I. Mani and M. Maybury. *Advances in Automatic Text Summarization.* MIT Press, ISBN 0-262-13359-8, 1999.

[18] D.R. Radev, H. Jing, and M. Budzikowska. Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *Summarization Workshop*, 2000.

[19] IBM Research Laboratory Tokyo. *Automatic Text Summarization.* www.trl.ibm.com/projects/langtran/abst_e.htm, last accessed on Oct. 9, 2002.

[20] Colorado State University. *Writing Guide: Interrater Reliability.* writing.colostate.edu/references/research/relval/com2a5.cfm, last accessed on Oct. 9, 2002.

[21] Y. Zhang, N. Zincir-Heywood, and E. Milios. World Wide Web site summarization. Technical Report CS-2002-8, Faculty of Computer Science, Dalhousie University, October 2002.