

WEBTALK: MINING WEBSITES FOR AUTOMATICALLY BUILDING DIALOG SYSTEMS

Junlan Feng Srinivas Bangalore Mazin Rahim

AT&T LABS RESEARCH

ABSTRACT

The task of creating customized spoken dialog applications has traditionally been known to be expensive, requiring significant resources and a certain level of expertise. This is clearly an obstacle in porting and scaling dialog systems especially those required for customer care and help desk applications. This paper describes WebTalk - a technique for automatically creating spoken and text-based customer-care dialog applications solely based on a company's website. Our goal is to create task oriented dialog services by automatically learning the task knowledge and mining information present on corporate websites. In this paper, we discuss the motivation and the feasibility of creating such a technique and present an overview of the main components of WebTalk. We address some of the challenges and present methods for evaluating such a system.

1. INTRODUCTION

Spoken natural language dialog systems provide a cost effective solution for call center and help desk automation. Some examples of successfully deployed spoken dialog systems include HMIHY [1], UBS Voice [4] and AOLByPhone[5]. There are several barriers that hinder the rapid portability of such systems to new services. The most significant challenge is minimizing human effort and knowledge required in building and maintaining dialog systems. The development of a conversational system involves a great deal of effort in task-specific data collection and design of handcrafted rules for the various components of a dialog system – a very expensive enterprise. As a result, only few of the large companies today take advantage in deploying spoken dialog systems in their customer service centers. Small companies shy away from exploiting the cost saving potential of such a service due to the enormous initial costs in developing a system and the ongoing need to maintain it. However, most companies spend significant time and effort into developing and maintaining a customer-friendly website. WebTalk is attempting to completely automate the process

of building spoken dialog systems by leveraging the wealth of information on websites. This technique will also enable dialog systems to be tightly synchronized with changes on the web.

Although, we are not aware of any literature that attempts to build conversational systems automatically based on the content of websites, there is a vast literature in the World Wide Web community on extraction of information from websites [7][9] and automated question-answering based on a collection of documents [15]. WebTalk incorporates some of the techniques present in this literature and extends them to provide a conversational interface to the underlying web content.

The paper is organized as follows. In Section 2, we discuss our categorization of website-driven dialog systems based on the methods of access. In Section 3, we describe the framework of WebTalk and its components. We discuss the issue of evaluation metrics in Section 4.

2. WEBSITE-DRIVEN DIALOG SYSTEMS

A corporate website is often designed and developed based on a thorough analysis of the market needs, competitive landscape, and functional requirements. In contrast to personal websites, a corporate website provides well-designed informative web content, a good navigation flow and customer-centered interactive applications (e.g. form filling). Furthermore, the web pages are systematically organized into subdirectories and are linked to each other through meaningful hyperlinks. The individual web pages use visual features to implicitly represent relations among units of information. All these features are professionally prepared for attracting visitors, serving customers and meeting the company's business needs.

Based on the nature of interactive content, we characterize website-driven dialog systems into three categories:

I. Information Retrieval Aid Dialog: In order to allow access to the large amounts of information present on a corporate website, most websites provide a local search engine which accepts keywords and returns a ranked list of links to relevant web pages. However, identifying the relevant piece of information from a list of

web pages is still a tedious and time consuming task. This problem is being addressed in the TREC question-answering track literature [14][16] [17]. The aim is to respond to users' natural language questions with exact answers rather than a list of documents. These tasks were restricted to answer factoid questions such as *In what year did Joe DiMaggio compile his 56-game hitting streak?* Expected answers are text snippets of 50 to 250 words from a pre-selected corpus. In reality, online customers are more likely to be looking for an answer similar in structure to the answers to Frequently-Asked-Questions (FAQ) or looking for an application (such as how to view and pay a bill) instead of a separate bit of information. Moreover, customers' requests may be out-of-task, illogical, or can't be answered with a text snippet in one turn.

In order to extend the question-answering paradigm, we propose Information Retrieval Dialog, which works through the following steps: (1) We apply question answering techniques to identify a natural information unit (IU) on web pages to provide a direct answer. (2) If that doesn't succeed, we retrieve relevant information units [13] and let the user know the system's knowledge state. (3) Otherwise, we provide context-based help prompts. IUs and help prompts are extracted or generated from the structure and content of websites.

II. Form-Filling Dialog: A company website often contains a number of forms which users fill out to accomplish a commercial activity or arrive at a customized information response page. Terms such as "hidden/invisible web" or "dynamic web" are used to describe the web content behind forms. With the dramatic increase of online businesses and online customer services, the hidden web plays an ever increasingly important role in web-based interactions. Search engines do not have a mechanism to access and index the hidden web. Each form has an associated web server application that is a result of time consuming analysis of the backend application database and related task logic. For example, a customer can simply type in his username and password for viewing or paying his bill. The response page, which visualizes the information as the user requests, is dynamically generated at the server side. Applying spoken dialog techniques to activate these forms and analyze the response pages will enable these online businesses to become available over the phone.

III. Table-based Dialog: Most traditional dialog systems are built using a dialog-oriented structured task knowledge representation. These knowledge sources permit the dialog system to conduct an intelligent dialog with the user. However, these knowledge sources are human authored and expensive to create. When dialog-oriented task knowledge can be successfully extracted from websites, we could significantly reduce or even eliminate the cost of authoring dialog systems. For

example, if products and services of a company, and their natural language definitions are automatically extracted in the form of tables, a table-based dialog system will be able to efficiently answer questions such as: "what products and services do you provide" or "what's the product in item 1". The tables are designed to store customer-centered structured task knowledge. We build on the results obtained from the web-oriented information extraction literature [6][8][9].

3. ARCHITECTURE OF WEBTALK

Figure 1 shows a schematic diagram of the six major technology components in WebTalk including Website Analyzer, Automatic Speech Recognizer (ASR), Language Understanding (LU), Dialog Manager (DM), Language Generation (LG), and Text-to-Speech (TTS) synthesizer. We expect this framework to support automation for building spoken dialog systems from a given website.

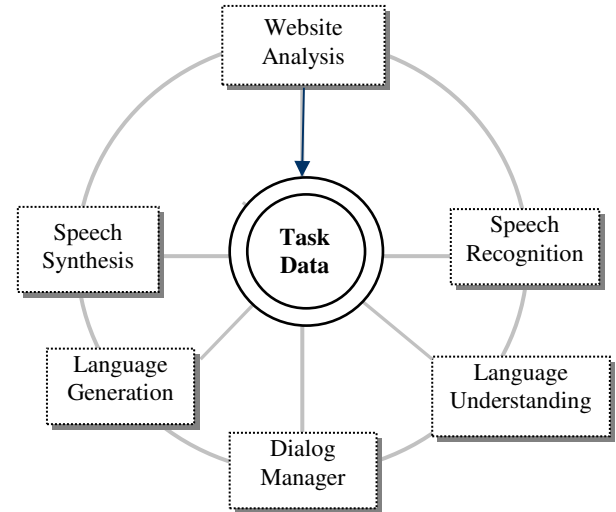


Figure 1: A schematic diagram of major technology components in WebTalk

3.1 Website Analyzer

Website Analyzer automatically constructs dialog oriented task knowledge from given websites. It takes websites as input and outputs task data in a number of formats, which are well suited for ASR, LU and DM to utilize. For instance, in order to achieve a task specific language model for ASR, Website Analyzer converts a website into a list of sentences. For configuring and providing data resources for DM and LU, Website Analyzer generates structured and semi-structured task data by exploiting the content and structure of the website. This task entails website structure analysis, webpage parsing and knowledge extraction. Figure 2 illustrates this process.

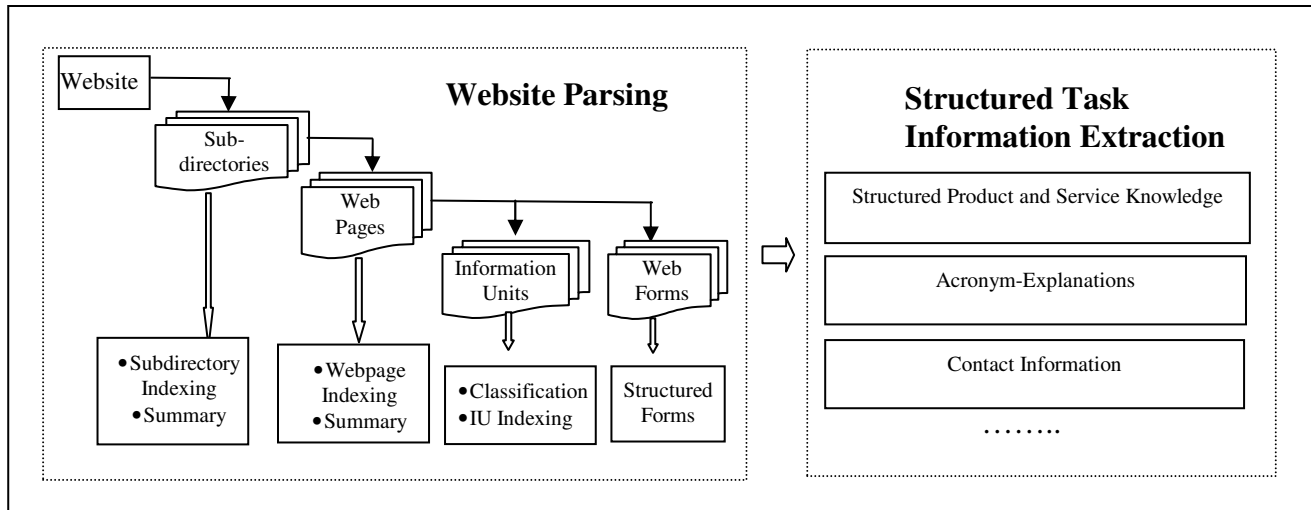


Figure 2: The diagram of Website Analyzer

A website often consists of text data that is distributed in many web pages. Parsing a website involves reorganizing the textual data into a representation which can be manipulated by a machine.

Website structure analysis: A website is more than just a collection of webpages. It is a systematic organization of web pages into subdirectories. Related web pages are weaved together through hyperlinks. WebTalk treats the website directory layout as a tree classified by task experts according to subject. Hyperlinks (or citations) are being actively used in web search engine, web crawlers, webpage classification and web-based question answering systems [11]. A hyperlink has three components: the source page where it is cited, the destination page and the anchor text describing the link. The assumption is that these anchor texts have greater discriminative and descriptive power than the text in the target document itself. Given a website, we collect all hyperlinks with semantically meaningful anchor texts and weight each anchor text proportionally to the frequency it is used and the importance of the pages, where the link is cited. These anchor texts plus page titles are used for building webpage indexing and sub-directories indexing, as shown in Figure 2. In addition, for each subdirectories and each page, a summary passage is generated. A directory summary consists of a list of anchor texts, which are descriptive and can cover most important content beneath the given subdirectory. A webpage summary is either a list of anchor texts or the page's first descriptive information unit. These summaries will be used as answers or help prompts by DM.

Webpage Parsing: As we observe in Figure 3, An HTML page may have a complicated layout. Its content is often multi-layered and often contains multiple information units (IU).

We define an information unit to be either a coherent topic area according to its content or a coherent functional area according to its behavior. An information unit is also often a visual block displayed in Internet browser. Humans can successfully follow visual cues and language cues to identify them. The task of Webpage Parser is to automatically detect IU boundaries and classify IUs into categories like MENU, LIST-ITEMS, QUESTION-ANSWER, TOPIC-EXPLANATION, EXPLANATION, FORM, and GARBAGE. IUs labeled as LIST-ITEMS, QUESTION-ANSWER, TOPIC-EXPLANATION, and EXPLANATION constitute the answer collection, from which DM chooses responses for users' requests. FORM is a special category. Figure 2 shows it separately. Forms are entrances leading to the hidden web as we discussed in section 1 and need further parsing into a structured format. MENU IUs are not candidate answers. But they are important for structured task knowledge extraction. Garbage IUs are short text segments like "copyright 2003", "click here", etc. An evaluation to this component will be detailed in section 4.



Figure 3: screen shot of www.att.com/

Information Extraction: Structured task knowledge extraction is prepared for the DM to answer users' questions precisely as in a table-based dialog system. Examples of information contained in the knowledge base include a list of product and service names, their definitions, their feature descriptions, their prices, and relationships among products and services. It also includes acronym-explanation pairs as well as direct contact information such as toll free phone numbers or email addresses. An abundant task knowledge base will evidently strengthen the system's intelligibility.

3.2 Speech Recognition

The speech recognizer we use in WebTalk is the AT&T Watson ASR engine [2]. In order to obtain high recognition performance, a task-oriented ASR component instead of a task-independent large vocabulary speech decoder component is strongly preferred. The challenge is we do not have access to any real examples about how human-machine interactions proceed within this task. Given this limitation, we have utilized written sentences from the website and structured task knowledge to derive a language model better suited to the task.

3.3 Language Understanding

Traditionally, LU is a rule-based component that converts natural language sentences into a semantic representation. For each new application, a set of handcrafted rules needs to be rebuilt upon task analysis. In WebTalk, language understanding is acquired to work automatically from website data. Hence, a data-driven language-understanding component is clearly needed. Within the above Webtalk framework, the language understanding module is able to:

- (1) Recognize the expectation of the input question, such as Definition, Price, Contact, Location, Signing up, Help, Greeting, Action and Others.
- (2) Identify the focus of the input question. Focuses are critical phrases, which might be product names or critical syntactic noun or verb phrases.

3.4 Dialog Manager

Dialog manager (DM) is the core of a dialog system and at the same time a component which involves the most handcrafted work such as predicting all possible dialog states, designing associated actions and responses for each state. This required dialog flow is not reachable in WebTalk. DM has to be data driven and needs to automatically work on the task data given by the Website Analyzer. At run time, the DM receives the output of the LU and generates a natural language response. The DM of WebTalk now is capable of:

- (1) Responding task independent users' requests, including greetings, commands (e.g. restart, repeat, speak slowly please, etc).
- (2) Using structured task knowledge to answer questions precisely and concisely.
- (3) Applying question-answering techniques to choose answers from the IU collection, webpage summaries or website subdirectory summaries.
- (4) Providing context-based help prompts when the system fails to fetch an answer by the above steps.

3.5 Language Generation

The language generation component of WebTalk is responsible for refining the text output of the DM to be dialog-style natural language prompts. The tasks include:

- (1) Presenting the answer more concisely. For this purpose, we developed an algorithm to perform query-relative summarization, which chooses sentences from a long answer based on relevance of sentences to the query and the degree of continuity between the chosen sentences.
- (2) Converting natural language text intended for display in a browser to a dialog-style prompts. For example: A text segment fetched by the DM – "Click here (hyperlinked to a URL: <http://www.att.com/spanich/>) for details." is replaced with "You can find more details on this page: <http://www.att.com/spanish/>". We handcrafted a set of general purpose replace rules to accomplish this task.
- (3) Translating table-based knowledge to natural language. Templates are manually created and used to realize a natural language explanation for certain types of questions. For example: "_stands for _" is a template for explaining the acronym such as "LNP stands for local number portability". Template-based language generation has been widely used in dialog systems [22].

3.6 Speech Synthesizer

WebTalk uses the AT&T TTS system [3] as its synthesizer.

4. EXPERIMENT

Figure 4 gives an actual dialog example between a user and WebTalk using www.att.com/local. The first turn is a general system prompt. "Hi this is _____. How may I help you?" is a sentence template; "AT&T local services" is automatically extracted from the website. The second system prompt is a simple discourse reply for continuing the conversation. The third turn includes a context-based system help prompt. The forth answer explains an acronym. The fifth one is an IU retrieved by referring to the dialog context. The sixth prompt is a task-independent

1. **WebTalk:** Hi, this is AT&T local services. How may I help you?
User: Hi.
2. **WebTalk:** Hi, how may I help you?
User: What can I say?
3. **WebTalk:** I can tell you about products and services, solutions, LNP, and local data services. How may I help you?
User: What's LNP, please?
4. **WebTalk:** LNP stands for local number portability.
User: Please tell me more about that.
5. **WebTalk:** You can download Handbooks and Guides from our website.
AT&T Business Services LNP Port Out Guide (PDF, 158 Kb)
AT&T Business Services Local Number Portability Procedures Handbook.pdf (PDF, 500 Kb)
User: Thanks
6. **WebTalk:** You're welcome. Is there something else I can help you with?
User: Do you have special solutions for small business customer?
7. **WebTalk:** A small and growing business has to prioritize expenditures and resource allocation. If your business requires 16-48 analog lines, and you don't want the expense of acquiring or operating an on-site PBX system, AT&T PrimePath NBX offers a peace-of-mind solution that gives your organization maximum flexibility.
User: What are your main products?
8. **WebTalk:** Our products and services include AT&T Centrex, Primeplex, Primecodes, T1 access, prime digital trunk, Prime NBX, Primeconnect, Primepath NBX, Primemail, Primelink, digital link, and Primexpress.

Figure 4: A dialog example with WebTalk, when working on www.att.com/local

common greeting. The seventh system response is constructed from summarizing a matched information unit. The last system reply comes from the structured product and service table.

Designing a study to evaluate the overall performance of WebTalk is a challenging task. Evaluating whether online customers prefer the natural language dialog mode of interaction over menu driven interaction or keyword search driven interactions is not the intention of our evaluation. Evidences supporting use of natural language dialog mode is a major research topic [20][21], and has received wide commercial interests (<http://www.internet.speech.com/>, <http://www.allie.att.com/>, <http://www.dejima.com/>). We wish to evaluate the performance of WebTalk, particularly in comparison with keyword search engine and handcrafted dialog systems. We are conducting experiments for these purposes. In the following, we present an evaluation of one component of WebTalk, namely the Webpage Parser.

Webpage Parser is designed for segmenting an html page into smaller information units (IU) and associating each IU with one category from MENU, LIST-ITEMS, QUESTION-ANSWER, TOPIC-EXPLANATION, EXPLANATION, FORM, and GARBAGE.

Each webpage is first represented as a table. Each row of the table corresponds to an atomic text segment of HTML source code, such as "7/13/2003". Each row has 12 cells, which correspond to various features of the text segment. (eg. color, background color, tags, parent tags, size, DOM-tree [10] [12] code, structure-code, position, link, server-action, client-action and text).

For boundary detection, the difference between two consecutive rows is transformed into a 186-dimension vector. For IU classification, each IU (a group of rows) is quantified into a 331-dimension vector. We apply support vector machine (SVM) [18] and IR-classification (IR-C) techniques [19] to detect the boundary between two IUs and classify each IU. Our approach provides a procedure for automatically creating general webpage parsing models instead of designing rules for every web resource.

Data Preparation: Two hundred pages have been randomly chosen from twenty websites selected from a variety of industrial sectors. We used one hundred and fifty pages as the training data and assign the remaining fifty pages for testing. Each page is labeled with a sequence of triples: {*from*, *to*, *Category*}. "*from*", "*to*" are ordering numbers of text segments. For an example, a page can be transcribed as follows:

0	4	MENU
5	20	MENU
21	25	TOPIC-EXPLANATION ...

"0 4 MENU" means that the first five text segments of this page constitute a menu.

Boundary Detection: For finding a better model to detect IU boundaries, we tested three binary classifiers including an IR classifier (IR-C), a SVM classifier, and a mixture model of SVM and a rule-based classifier. Both the IR classifier and the SVM classifier are trained using the training data. Table 1 shows the result when evaluating these classifiers on the testing data. From this table, we can find that SVM is an efficient model for solving this issue.

	Recall (Hit Rate)	Precision	Classification Accuracy
IR-C	59.0%	93.5%	90.5%
SVM	77.9%	95.5%	94.5%
SVM+Rules	84.8%	95.5%	96.4%

Table 1: The performance of boundary detection

Recall : the probability that a vector x labeled as a boundary is classified correctly.; Precision : the probability that a vector x classified as a boundary is indeed a boundary

IU classification: Table 2 gives the accuracy of IU classification using the IR-C and a set of rules.

	Classification Accuracy
IR-C	74.6%
IR-C+Rules	91.6%

Table 2: The performance of IU classification

5. SUMMARY

This paper describes WebTalk, a general framework for automatically building spoken dialog systems from mining the content of websites. The goal is to enable companies, which have already set up their websites, to extend their customer service with a spoken dialog interface whether over the phone or through the Internet.

In this paper, we addressed the challenges confronting each WebTalk component, presented our efforts on implementing Website Analyzer, Language understanding, Language Generation, and Dialog manager, and provided the evaluation of Webpage Parser – an essential component of WebTalk. Detailed evaluation of the entire system will be reported in a future publication.

6. REFERENCES

- [1] A.L. Gorin, B.A. Parker, R.M. Sachs and J.G. Wilpon , "How May I Help You", Proc. IVTTA, Basking Ridge, NJ, Oct. 1996
- [2] R.D Sharp; E Bocchieri; C.Castillo; S. Parthasarathy; C Rath; M. Riley; and J.Rowland. The Watson speech recognition engine, In Proc. Int. Conf. Acoustic, Speech , Signal Processing; pages 4065-4068; 1997
- [3] M.Beutnagel; A. Conkie ; J. Schroeter; Y. Stylianon; and A. Syral. The AT&T next-generation. In Joint Meeting of ASA; EAA and DAGA; 1999.
- [4] http://www.nuance.com/corp/customers/cases_studies/ubs.html
- [5] <http://www.speechworks.com/>
- [6] N. Kushmerick, D. Weld, R. Doorenbos, Wrapper induction for information extraction, in: Proc. IJCAI-97, Nagoya, Japan, 1997, pp. 729–735.
- [7] Line Eikvil: «Information Extraction from World Wide Web - A Survey». Rapport Nr. 945, July, 1999. ISBN 82-539-0429-0.
- [8] William W. Cohen, Matthew Hurst, Lee S. Jensen, "A Flexible Learning System for Wrapping Tables and Lists in

HTML Documents", WWW2002, May 7-11, 2002, Honolulu, Hawaii, USA, pp232-241

[9] A. Maedche, G. Neumann and S. Staab, Bootstrapping an Ontology- Based Information Extraction System. Studies in Fuzziness and Soft Computing, editor J. Kacprzyk. INTELLIGENT EXPLORATION OF THE WEB, P.S. Szczepaniak, J. Segovia, J. Kacprzyk, L.A. Zadeh, Springer 2002/01/01

[10] Shipeng Yu, Deng Cai, Ji-Rong Wen, Wei-Ying Ma, "Improving Pseudo-Relevance Feedback in Web Information Retrieval Using Web Page Segmentation", proceeding of WWW2003, Budapest, HUNGARY, pp Pages: 11 - 18

[11] Eric J. Glover, Kostas Tsioutsoulis, Steve Lawrence, David M. Pennock, Gary W.Flake, "Using Web Structure for Classifying and Describing Web Pages", WWW2002, May 7-11, 2002, Honolulu, Hawaii, USA, pp 562-569

[12] Gerald Penn, Jianying Hu, Hengbin lu, Ryan McDonald, "Flexible Web document Analysis for Delivery to Narrow-bandwidth Devices", ICDAR2001

[13] S.Lawrence, C.I. Giles. Searching the World Wide Web, Science magazine,v.280, pp.98-100, April 1998

[14] Zhang, W.S. Lee, "Web Based Pattern Mining and Matching Approach to Question Answering", in the Proceedings of TREC 2002

[15] Ellen M. Voorhees, National Institute of Standards and Technology, "Overview of TREC 2002"

[16] C.L.A. Clarke, G.V. Cormack, T.R. Lynam, C.M. Li, G.L. McLearn,"Web Reinforced Question Answering (MultiTest Experiments for TREC 2001", in the Proceedings of TREC 2001, page 673

[17] David Pinto, Michael Branstain, Ryan Coleman, WBruce Croft, Matthew King, "QuASM: A System for Question Answering Using Semi-Structured Data", JCDL'02, July 13-17, 2002, Portland, Oregon, USA

[18] Vladimir N. Vapnik, The Statistcal Learning Theory. Springer, 1998.

[19] Rainer Hoch, "Using IR Techniques for Text Classification in Document Analysis, Proceedings of SIGIR-94 ", 17th ACM International Conference on Research and Development in Information Retrieval.

[20] Joyce Chai, Jimmy Lin, et.al, "Comparative Evaluation of a Natural Language Dialog Based System and a Menu Driven System for Information Access: a Case Study", in proceeding of the International Conference on Multimedia Information Retrieval(RIAO 2000)

[21] Mashiro Araki, Tasuku Ono, Kiyoshi Usda, Tasuhisa NHMI, "An Automatic Dialogue System Generator from the Internet Information Contents", Euro speech 2001

[22] Jyag & Idey, "A Template-Based Generator and Its Authoring Tool", In the Proceedings of The 18th National Conference on Artificial Intelligence (AAAI 2002), Edmonton, Canada, July 2002.