Question-Answering in WebTalk: An Evaluation Study

Junlan Feng, Srinivas Bangalore, Mazin Rahim

AT&T LABS RESEARCH
Florham Park, NJ 07932, USA
{junlan, srini, mazin}@research.att.com

Abstract
WebTalk is an intelligent agent that automatically mines company websites to create interactive spoken and chat-based dialog systems. In this paper, we briefly describe the technologies that WebTalk utilizes for extracting task knowledge from websites and answering questions specified in natural language. We elaborate on an evaluation of the question answering component of WebTalk. In particular, we compare this component with a deployed hand-crafted question answering agent. A detailed evaluation analysis is presented.

1. Introduction
The development of a conversational system typically involves a great deal of effort from collecting task-specific data to designing rules for the various components of a dialog system – a very expensive enterprise. WebTalk [1] attempts to automatically create spoken and chat-based customer-care dialog applications solely based on a given company website. A WebTalk system consists of six major technology components including Website-based Knowledge Mining (KM), Automatic Speech Recognizer (ASR), Question Answering (QA), Dialog Manager (DM), Language Generation (LG), and Text-to-Speech (TTS) synthesizer. The aim is to automatically create and maintain the underlying models and data for each of these components by exploiting the content and structure of the given website. In previous work [1], we described these components and discussed the associated technology challenges. This paper addresses the issue of WebTalk evaluation. In general, evaluating spoken or chat-based dialog systems is a major research challenge. Most efforts have typically focused on specific applications [2]. Unlike most traditional dialog systems that are configured by a hand-crafted dialog flow and designed for completing tens to hundreds of actions, a WebTalk dialog application is controlled by information provided on a website resulting in an unstructured and extensive dialog flow. This naturally presents a bigger challenge when attempting to evaluate WebTalk.

As a step towards evaluating the entire WebTalk, this paper focuses on evaluating one of its components, namely the QA. Section 2 describes the KM and the QA components. Section 3 presents an evaluation framework and a comparative study with a deployed hand-crafted question-answering agent. Conclusions are presented in Section 4.

2. WebTalk
Figure 1 shows an example of WebTalk dialog when instantiated on a telecom company website. There are two dialog turns in this example. WebTalk answers the first user input as a QA system does. For responding the second request, WebTalk associates “this feature” with “Locate Me” based dialog context information it keeps. Evaluating a complete WebTalk dialog system is an ongoing research effort. In this section, we focus on how WebTalk mines a website, analyses a question and provides answers.

User: How may I help you?
WebTalk: How may I help you?
User: What's Locate me?
WebTalk: LOCATE ME: Set up your service so callers can find you at other phone numbers.
- so you don't miss your important calls.
User: How can I activate this feature?
WebTalk: From your Personal Call Manager website click "Locate Me" to go to the Locate Me setup page, where you can turn it on and enter the other numbers where you can be reached.

Figure 1: An example dialog with WebTalk

Question answering by itself has become an increasingly popular topic [3]. A QA system provides users precise answers instead of giving a long list of relevant documents. Most modern question answering systems provide short answers to factoid questions, such as “which river is the biggest river in the world?”. Using the World Wide Web or a pre-collected text corpus. In the TREC question answering track[3], participating systems like LCCmain2002, ExactAnswer, IBMPQSQACYC, and BBN2002C have been designed to answer questions based on a pre-selected newswire corpus. Web-based question systems such as Start/OmniBase, MULDER, AskMSR, and AskJeeves use enormous amounts of web data as a large repository of answers. Both web-based and corpus-based question answering techniques have evolved over the years. Some of the existing systems have achieved high-quality results.

However, bridging the gap between a question and its expected answer still remains a great challenge. In reality, there are a variety of question types, each of which needs to be treated differently, and the knowledge required to appropriately connect a question with an answer is usually quite extensive. For instance, for a question expressed as “What equipments do I need for signing up to this service?” an eligible answer might be: “This service requires the following hardware: …”. To validate this answer, the system needs to know the relevance between “equipments” and “hardware” as well as “need” and “require”. Web-based QA systems face more challenges such as parsing unstructured
web pages and dealing with redundancy, recency, and reliability of web page content. WebTalk question answering is targeted to answer customer care questions using a business website. It looks for precise answers similar in structure to the answers to Frequently-Asked-Questions (FAQ). In the following, we first discuss the characteristics of a business website and customer needs, and then describe our approaches to mining websites and answering questions.

2.1. Characteristics of a business website

A business website is professionally prepared for attracting visitors, serving customers and meeting the company’s business needs. In contrast to personal websites, a business 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 cross referred through meaningful hyperlinks. The individual web pages often have an implicit structure representing relations among units of information. Content composed on a company website is more reliable than that on the World Wide Web. A dialog system that leverages a business website, including the website structure, navigation flows, web page structures, and all the available webpage contents which are structured, semi-structured or unstructured, has the potential to help a wide range of customer requests.

2.2. Customer requests

We classify customer requests into five categories - Generic Information Request, Problem Reporting, Factoid Questions, Transaction Request, and Information Search. Generic Information Request comprises the questions concerning the general corporate product and service information, such as “How is the Inkjet printer different than a laser printer?” These questions expect a concise explanation passage to be the response. Problem Reporting is the most frequently asked question type. Examples are like “I can not log in”, “Why don’t I always receive e-mail notification of my online bill?” These questions expect an interactive dialog, Alternative responses can be topic-related instructions. Factoid questions might expect a commonly used entities like Price, Date, Time, Phone number, Email-address, Mailing-Address, Web-Address or broad facts, such as “what equipments do I need?” Transaction Request corresponds to questions that ask for completing a transaction. Typical examples are: “I want to receive my bill by mail.”, “Sign up for a new account”, “make a payment.” For a customer care website, some of these requests can be self serviced through HTML forms. Information Search groups the questions which look for relevant information but don’t have specific intentions such as “connectivity charge” and “technical help for internet”.

2.3. Knowledge mining

A website is an organized collection of web pages. First, KM captures the website architecture by following hyperlinks and the website directory hierarchy and represents this structured information in XML. Second, each individual web page is parsed into a sequence of information units, which are defined as a coherent topic area according to its content or a coherent functional area according to its behavior. WebTalk classifies these information units into one of the following categories: menu, form, table-data, FAQ-answer, headline-content, bulletined list, ordinary passages, and garbage. Each category is represented by an XML entity. These patterns are considered as the meta data of each unit. Third, KM extracts structured task knowledge, including products and services of the company, properties of these products and services, corporate contact information, as well as acronyms the website uses. WebTalk question answering component uses these data as the answer repository.

2.4. WebTalk question answering

There are four steps in the WebTalk question answering process: question classification, query formulation, answer retrieval, and answer validation.

Question Classification: A user’s question is classified into the five categories listed Section 2.2. Factoid questions are further classified into Location, Phone Number, Email-Address, Name, Mailing-Address, Web-Address and Broad Facts. Question classification also decides if the question is related to or requesting for customer personal data. Currently, question classification is a rule-based component in WebTalk.

Query Formulation: Query formulation generates a set of query terms from a natural language question. In the first step, phrases are detected based on the WordNet compound word dictionary [4] and a website specific phrase table which is extracted from the given website. For example, “long distance” is an atomic semantic unit for the telecom industry; “turn on” is a general phrase. In the second step, we extract the basic query terms by removing stop words in the question and normalize each term to its base form using a morphological processor [4], for instance, “differences” is converted to “difference”. In the third step, we expand the base query terms using the task knowledge and the WordNet lexical reference system [4]. Task dependent query expansion extends an acronym to the phrase it stands for and expands a query term with its equivalent representations. For instance, “Speed-dial” and “Click-dial” refer to same service feature in a particular task. Task independent query expansion involves expanding an adjective term with its noun form (e.g. “difference” is the noun form of “different”), expanding a noun term with the associated adjective values and expanding terms with its most frequently used synonyms (e.g. “utilize” is a synonym of “use”). Through these steps, a question like “How can I use the speed-dial feature?” is represented with a set of terms – [(use I utilize), [speed-dial I click-dial], feature].

Answer Retrieval: This stage retrieves answer candidates from the website data and ranks them based on an answer confidence score. Our retrieval approach improves on the standard tf.idf Information Retrieval (IR) framework, which represents queries and documents with term vectors and measures query-document relevance with vector cosine angel. Term vectors are weighted through tf.idf weighting mechanism [5]. We extend this measure by (i) emphasizing the terms the user stresses in the question; (ii) taking into account the meta data of answer resources through assigning higher weights to the terms used in content-headlines, table headlines, and hyperlink anchor texts; (iii) rewarding earlier question focus[6] match, i.e., when customers ask a question, they expect the
response to seize the focus of their question immediately and the earlier the focus is stressed, the more responsive they judge the answer is; (iv) rewarding QA structure similarity. The structure is represented by stop word distance between terms. An example representation, “W1 [3] W2 [0] W3” means there are three stop words between the first term W1 and the second term W2, and W2 and W3 are consecutive. The distance between W1 and W2 is quantified as 3. When an answer includes two terms that appear in the question, two stop word distances between those terms are captured respectively for the question and the answer. A best QA structure match is assumed that these two distances are equivalent.

**Answer Validation:** This component decides the final response. It checks the ranked list of answer candidates and outputs the first one which meets the following conditions: (1) The answer confidence score is above a predefined threshold; (2) The answer contains the products or services referred in the question; (3) The answer matches the question type. If none of the answer candidates meet these conditions, the system returns the string “NIL”.

### 3. Evaluation

In this section, we elaborate on an evaluation of our website-based question answering techniques.

#### 3.1. Evaluation results

In order to test WebTalk with real-world customer questions, we conducted an evaluation using a telecommunication website, where we have access to the associated customer requests. In particular, we compare the QA component of WebTalk with a deployed question answering agent that is manually designed for that particular website and allows customers to input their question in natural language. We refer to this agent as HandQA. HandQA is built on a corporate specific ontology database and 1600 predetermined FAQ-answer pairs. For every submitted customer question, HandQA chooses one of these 1600 answers as the response. WebTalk did not have access to this pool of answers and the ontology database. WebTalk provided answers based on information scattered across the website. The system performance of HandQA can be viewed as the upper limit that a fully automated system like WebTalk can achieve.

For this experiment, we used two test sets. Set-A includes 800 randomly picked real customer questions. Set-B includes 800 FAQs that is a subset of the 1600 FAQ pool from HandQA. We fed Set-A and Set-B to both WebTalk and HandQA. Systems were limited to one response per question. WebTalk returned 1423 answers and 177 “NIL” responses. HandQA provided 1493 answers and rejected 127 questions. In total, we achieved 2916 QA pairs from the two systems. Obviously, for Set-B, HandQA returned the pre-crafted answers.

We designed a web-based interface for human assessors to log in and evaluate the quality of these answers. QAs are randomly picked from the 2916 QA pair set and presented to assessors to rate. The assessor doesn’t know which system the given answer is from. Assessors can evaluate as many QAs as they volunteer to do. An assessor doesn’t evaluate the same QA twice. Each QA is rated on a scale from 1(Bad) to 5(Excellent) as defined as below:

<table>
<thead>
<tr>
<th>Score</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Bad</td>
<td>The answer is completely irrelevant.</td>
</tr>
<tr>
<td>(2) Poor</td>
<td>The answer is somewhat related to the question but does not answer it.</td>
</tr>
<tr>
<td>(3) Fair</td>
<td>The answer is helpful but does not directly answer the question.</td>
</tr>
<tr>
<td>(4) Good</td>
<td>The answer contains a direct answer to the question.</td>
</tr>
<tr>
<td>(5) Excellent</td>
<td>The answer exactly provides what the question requests.</td>
</tr>
</tbody>
</table>

#### 3.2. Analysis of the evaluation

A set of 26 assessors participated in this experiment, submitted 1896 ratings and evaluated 1441 QA pairs. 387 QAs received multiple ratings. A QA pair receives more than one rating when it is evaluated by multiple assessors.

Table 2 gives the evaluation results for Set-A. The table shows the average answer rating, the percentage of ratings which are equal to or greater than 3 as well as the percentage of ratings which are equal to or greater than 4 for both WebTalk and HandQA, which seem to be generally comparable. For example, the average answer rating of WebTalk 2.35 is 88% of the average answer rating of HandQA 2.66. These results are very encouraging given that WebTalk was created fully automatically and had no access to any extra business specific data except for the website.

Table 3 illustrates how WebTalk performed on Set-B. Answers from HandQA for Set-B are predefined by customer care experts. Therefore, the performance of HandQA on Set-B is the upper limit that an automated QA system can achieve. The contrast between WebTalk and HandQA in Table 3 shows how far the question answering performance of WebTalk is from that of human. The average answer rating of WebTalk on Set-B 2.5 is 56% of that of HandQA 4.46.
retrieval sub-components. We assume that these sub-components are independent of each other and each of them characterizes a QA feature \( f_j(q,a,W) \), where \( W \) is the corpus from which the answer \( a \) is retrieved for a given question \( q \), and \( i = 1, ..., T \), where \( T \) is the size of the feature set. One example feature is the tf.idf similarity previously described in Section 2.4. WebTalk combines the feature scores through a function \( \pi(f_1, ..., f_T) \) to calculate the answer confidence score \( c \):

\[
    c = \pi(f_1, ..., f_T) \tag{3.1}
\]

The true quality of an answer is rated by human and denoted as \( e \). Given a set of features, we look for a linear combination function \( \pi^* \) which maximizes the correlation between \( e \) and \( c \):

\[
    \pi^* = \arg\max_\pi R(c,e) \tag{3.2}
\]

\( R \) denotes the correlation coefficient of two variables. \( \pi \) is represented as:

\[
    \pi(f_1, ..., f_T) = a_0 + a_1 f_1 + ... + a_T f_T \tag{3.3}
\]

In this experiment, we used 11 QA features, i.e., \( T=11 \). We represent the 1441 evaluated QAs with:

\[
    e_i = a_0 + a_1 f_{i1} + ... + a_T f_{iT}, \quad i = 1, ..., 1441 \tag{3.4}
\]

Unknown coefficients \( a_0, a_1, ..., a_T \) were solved by performing a Linear Regression fit [7]. \( e_i \) is the average assessor rating on the \( i \)th QA. To test this model, we instantiated these coefficients in equation (3.3) and achieved the answer confidence score \( c^* \). The correlation between \( c^* \) and \( e \) \( R(c^*, e) \) was calculated as 0.58. The highest individual feature correlation \( R(f_j, e) \) was 0.39, where the feature \( f_j \) is the QA structure similarity that was explained in Section 2.4. This indicates that \( c^* \), the predicted answer confidence score using the Linear Regression Fitting Model, provides better correlation with assessor rating compared to individual features.

The second experiment we conducted is to apply Support Vector Machines (SVMs) to predict if an answer is a helpful answer (\( e \geq 3 \)) or an irresponsible answer (\( e \leq 3 \)) as well as if the answer is a good answer (\( e \geq 4 \) or not a good answer. Assessor ratings \( e_i \) are binarized as below:

**Classifier-I:**

\[
    e'_i = \begin{cases} 
    1 & e_i \geq 3 \\
    0 & \text{otherwise}
    \end{cases}
\]

**Classifier-II:**

\[
    e'_i = \begin{cases} 
    1 & e_i \geq 4 \\
    0 & \text{otherwise}
    \end{cases}
\]

We trained and tested these two binary SVM [8] classifiers, Classifier-I and Classifier-II, using 10 fold cross-validation [9]. Table 4 gives the classification accuracies with 11 features. Classifier-I achieved 69.3% classification accuracy. Classifier-II achieved 70.1% accuracy. This result shows that WebTalk can discriminate good answers from poor answers with 70% accuracy by using SVM.

### Table 4: Feature analysis using SVM

<table>
<thead>
<tr>
<th>Method</th>
<th>Features</th>
<th>Classifier-I Accuracy (%)</th>
<th>Classifier-II Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear SVM</td>
<td>( X = {f_1, ..., f_{11}} )</td>
<td>69.3</td>
<td>70.1</td>
</tr>
</tbody>
</table>

### 4. Conclusions

WebTalk is a framework for automatically building customer care dialog systems from mining business websites. In this paper, we briefly described the techniques used for WebTalk question answering and established a framework for evaluating this component. Experimental results show:

- The question answering performance of WebTalk is 88% of an in-domain hand-crafted agent—HandQA.
- Combining features using a Linear Regression Fitting Model provides better correlation with assessor ratings.
- A binary SVM classifier built on 11 features achieved 70% accuracy in discriminating good from poor answers.

### 5. References