INTERNATIONAL JOURNAL OF SPEECH TECHNOLOGY 4, 285–295, 2001

The Role of a Natural Language Conversational Interface in Online Sales: A Case Study

JOYCE CHAI
IBM T. J. Watson Research Center, 30 Saw Mill River Rd., Hawthorne, NY 10523, USA
jchai@us.ibm.com

JIMMY LIN
MIT Artificial Intelligence Lab, 545 Technology Square, Cambridge, MA 02139, USA
jimmy@mit.edu

WLODEK ZADROZNY, YIMING YE, MARGO STYS-BUDZIKOWSKA, VERONIKA HORVATH, NANDA KAMBHATLA AND CATHERINE WOLF
IBM T. J. Watson Research Center, 30 Saw Mill River Rd., Hawthorne, NY 10523, USA
wlodz@us.ibm.com
yiming@us.ibm.com
sm1@us.ibm.com
veronika@us.ibm.com
nanda@us.ibm.com
cwolf@us.ibm.com

Received August 9, 2000; Revised March 16, 2001

Abstract. This paper describes the evaluation of a natural language dialog-based navigation system (HappyAssistant) that helps users access e-commerce sites to find relevant information about products and services. The prototype system leverages technologies in natural language processing and human-computer interaction to create a faster and more intuitive way of interacting with websites, especially for less experienced users. The result of a comparative study shows that users prefer the natural language-enabled navigation two to one over the menu driven navigation. In addition, the study confirmed the efficiency of using natural language dialog in terms of the number of clicks and the amount of time required to obtain the relevant information. In the case study, as compared to the menu driven system, the average number of clicks used in the natural language system was reduced by 63.2% and the average time was reduced by 33.3%.

Keywords: conversational interfaces, human-computer interaction, dialog systems, natural language processing

1. Introduction

With the emergence of e-commerce (Aggarwal et al., 1998; Muller and Pischel, 1999), successful information access on e-commerce websites that accommodates both customer needs and business requirements becomes essential. Menu driven navigation and keyword search provided by most commercial sites have tremendous limitations. The menu driven approach is likely to overwhelm users and frustrate them with lengthy interactions. A recent study shows that the user's interest in a particular site decreases exponentially with the increase in the number of mouse clicks (Huberman et al., 1998). Therefore, shortening the interaction path to provide useful information becomes important. On the other hand, keyword search engines usually require users to know domain-specific jargon. Keywords are not only unable to describe the user's intention precisely, but more importantly, they might not match words used in the catalog or documents. Furthermore, the keyword search lacks understanding of semantic meanings of the search words or phrases. For example, keyword search cannot understand that "summer dress" should be looked up in women's clothing under "dresses", whereas "dress shirt" most likely in men's under "shirts". A search for "shirt" can reveal dozens or even hundreds of items, which is useless for somebody who has a specific style and pattern in mind. Moreover, search engines do not accommodate business logic, e.g., a prohibition against displaying cheap earrings with more expensive ones. The solution to these problems lies, in our opinion, in centering e-commerce websites on natural language (and multimodal) dialog. This claim is supported by results of a recent study we performed, and which will be presented in this paper.

Natural language dialog has been used in many areas, such as for call-center/routing applications (Carpenter and Chu-Carroll, 1998; Chu-Carroll and Carpenter, 1998), e-mail routing (Walker et al., 1998), information retrieval and database access (Androutsopoulos and Ritchie, 1995), and for telephony banking (Zadrozny et al., 1998). The integration of natural language dialog with an e-commerce environment is a novel feature of our system. Our work goes beyond the "natural language interface" features of websites such as www.askjeeves.com and www.neuromedia.com, which work in a question-answer mode and do not use dialog. This is a crucial difference. When searching e-commerce sites, users often do not know where to find information, or how to specify a request although they have targets in their minds. Sometimes they have only vague or no targets in minds (Saito and Ohmura, 1998). Thus they need to formulate or revise their request based on additional information, which can be provided in a dialog. Our study shows that natural language dialog is a very effective medium for negotiating such contexts by understanding the user's requests/intentions and providing help/advice/recommendations to the user.

Furthermore, information access on e-commerce sites is different from traditional keyword search or information retrieval where the business decisions are not supported. However, for a business to successfully operate online, certain business strategies should be enforced to facilitate the search. For example, very expensive earrings should not be shown together with cheap necklaces when the user is searching for matching earrings and necklaces.

Having those issues in mind, we built a proof-of-concept system and conducted a comparative user study to test our hypothesis. The system allows customers to make requests in natural language and be directed towards appropriate web pages that sell the product or provide the service. Users can type what they are looking for in natural language. The system identifies and understands key concepts from the user's input. Then by applying the user's concepts to business rules, the system either will display the web page that satisfies the user's requests or initiate a dialog with the user to ask for additional information and clarify the request. We conducted a user study to evaluate the natural language dialog based system, particularly, in comparison with a menu driven system. The result shows that users prefer the natural language dialog mode of interaction two to one over the menu driven interaction. The preference is stronger for less experienced internet users.

In this paper, we will first describe the natural language dialog based system (HappyAssistant). Then we will report the results from the comparative evaluation of this system and a menu driven system. Finally, we will discuss what we have learned from the study and propose future work.

2. The HappyAssistant

2.1. System Architecture

The architecture of the system supports multimodal dialog. For this case study, the prototype system was implemented for textual input, with the option of browsing non-textual information. However, our architecture is designed to support inputs from different channels and modalities, including keyboard input and output, speech input and output over a telephone, speech input and output over a microphone, mouse input, pointing device input, and dataglove. The system consists of three major modules: Presentation Manager (PM), Dialog Manager (DM) and Action Manager (AM). The Presentation Manager is responsible for separating content from the presentation mode. In the prototype system, the Presentation Manager employs a natural

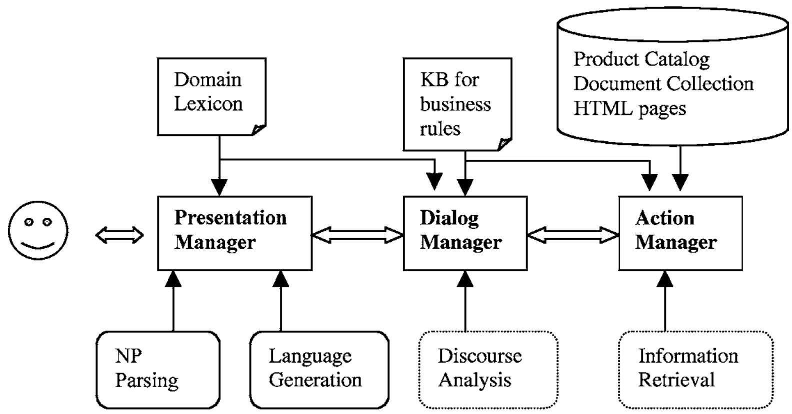

Figure 1. System architecture.

language parser to transform the user's natural language query into a logical form, and sends the logical form to the Dialog Manager. It is also responsible for obtaining the system's response from the Dialog Manager and presenting it to the user. The Dialog Manager is responsible for determining the specific action(s) requested by the user and filling the parameters (e.g., the attributes of the computers in which users are interested) of the identified action by way of a dialog with the user. The Knowledge Base for business rules specifies the translation from user requests to action plans for the Action Manager to satisfy the requests, for example, retrieving information about particular computer models from the product catalog. The architecture is shown in Fig. 1.

2.2. *Domain Knowledge*

World knowledge and/or domain knowledge is an important component in all dialog systems. Various representations and approaches have been used. TRAINS uses a domain plan reasoner that maintains a representation of the state of the world and reasons about the domain (e.g., gives suggestions about train routes) (Ferguson et al., 1996). GALAXY's conceptual model consists of semantic classes and relations among them. The application data is being stored on domain servers that contain more specific domain knowledge used for the interpretation of semantic frames (Seneff et al., 1996). RAILTEL implements a conceptual model (semantic frames) and simple inference rules. The domain knowledge consists of two kinds of rules. Default value rules define defaults, and interpretative rules are responsible for transformations of imprecise qualitative values into quantitative specifications (Bennacef et al., 1996). We use XML-based concepts and rules (Bray et al., 1998; Radev et al., 1999) to represent and manage the domain knowledge. The Domain Lexicon is an external dictionary file that maps keywords onto concepts. For example:

```
<ENTRY NORMAL_FORM="affordable-concept"
       QUESTION ="What are your financial
         constraints?"
       BACKOFF_QUESTION="Is affordable
         price important to you?">
  <WORD>affordable</WORD>
  <WORD>cheap</WORD>
  <WORD>inexpensive</WORD>
  <WORD>reasonably priced</WORD>
</ENTRY>
```

The <WORD> elements are keywords that trigger the concept, and the QUESTION attribute is a natural language question designed to elicit that particular concept and related ones. Concepts can be as concrete as a category of computer, e.g., desktop-concept, or as abstract as an idea, e.g., performance-and-value.

A business rule consists of a list of concepts together with some metadata about the target product or service:

```
<RULE>
  <CONCEPT-LIST>
    <CONCEPT>notebook-concept</CONCEPT>
    <CONCEPT>high-tech-concept</CONCEPT>
    <CONCEPT>fast-concept</CONCEPT>
  </CONCEPT-LIST>
  <WEIGHT>0.9</WEIGHT>
```

```
  <DESCRIPTION>The IBM ThinkPad 770
    </DESCRIPTION>
  <LONG_DESCRIPTION><![CDATA[
      <P><IMG SRC=http://...>
      <P>The ThinkPad 770 is IBM's top
          of the line laptop, offering the
          ultimate in performance and
          display.
    ]]></LONG_DESCRIPTION>
  <URL><![CDATA[http://...]]></URL>
</RULE>
```

The <WEIGHT> element is designed to reflect the importance of this particular rule from the business aspect. In the case study, we manually set this value. However, in a real business setting, this weight should reflect customer demands, business supplies, market competition, etc. For example, if the company wants to promote the ThinkPad 600 product, a business rule that translates a set of concepts to the ThinkPad 600 should have a higher weight than business rules that translate the concepts to other products.

Knowledge management is a key issue in information systems. The closed-world approach has tremendous limitations and often fails when the task becomes complex and when the application environment evolves. Therefore, a dynamic updating procedure that reflects the evolved state and knowledge of the world, the domain and the business is essential. Moreover, timeless and semi-automated (if not fully automated) updating is desirable. Although in the proof-of-concept system the Domain Lexicon and the Knowledge Base for business rules were created manually, we have developed a set of tools and processes to automate knowledge maintenance in the next generation of the system (Chai et al., 2001).

2.3. *The Role of Natural Language Dialog*

The HappyAssistant algorithm capitalizes on a major advantage of natural language, which is the ability to ask very general questions and elicit rich descriptive responses from the user (i.e., the ability to choose from a very large set of possible attributes). Obtaining a larger set of user requirements through the dialog shortens the interaction length and thus improves its quality. The dialog is initiated by matching concepts from the user's query to business rules. If a match is found, then a web page associated with that rule is presented to the user. Otherwise, based on the weights of partially matched rules, the system finds the most important missing concept and asks a question to elicit descriptive responses from the user.

More specifically, each rule has a rank. Rank is defined as the number of concepts in a rule that has *not* been extracted from user queries in a particular session. For example, if a rule requires three concepts to trigger, and only two of those concepts have been identified, then this particular rule is assigned a rank of one. The initial matching algorithm iterates through all the rules in the knowledge base, and calculates the rank of each rule. The presence of any rules with rank zero indicates the triggering of that rule, i.e., that particular product or service matches the user description. In this case, a detailed web page regarding that item is displayed in a separate browser window. If there are multiple rule triggerings, the rule with the highest weight is selected. If there are no rules of rank zero, then additional refinement is needed to recommend a suitable item. HappyAssistant chooses from the rules of the lowest rank and the highest weight, and from that finds a concept that has not yet been identified. The system retrieves the natural language question associated with that concept from the Domain Lexicon and poses it to the user, prompting for a response. Simultaneously, all items associated with partially matched rules (up to a certain adjustable upper limit) are offered to the user as 'items of interest.' The user has the option either to answer the question posed or browse through the list of relevant items.

The natural language question posed by the HappyAssistant for each concept is not merely intended to affirm or deny that concept, but to further elicit descriptive responses from the user. This allows the system to collect more information about user needs. For example, the question associated with the affordable-concept (people who are looking for an affordable computer) might be "What are your financial constraints?" This tactic would allow the HappyAssistant to differentiate among similar concepts such as value-performance-combination (i.e., a mixture of value and performance) or unconstrained-finances (i.e., don't care about price) in a single exchange. After the concepts are extracted from the user reply, the partially matched rules will be re-matched again. This refinement process of question and answer repeats until the system can recommend an item to the user or until all possible items have been eliminated, in which case, a graceful exit message is displayed.

To assist users who have trouble describing their requirements, the HappyAssistant implements a back-off mechanism that rephrases the question in a more specific way. For example, if the question "What are your financial constraints" does not elicit any recognizable response, the system would proceed to ask a different question, "Is affordable price important to you?" In general, this dialog-based information access task is accomplished by implementing a hybrid forward and backward chaining rule-based system. An initial description is gathered from the user, creating a set of currently active rules, which is refined by subsequent query and response. Because the rules operate on concepts, they can be developed independently apart from the language analysis section of the system.

In this proof-of-concept system, the dialog management is rather naïve. In our next generation system, we implemented a state-based dialog manager that applies mixed initiative dialog strategies to arrange different follow-ups (Chai et al., 2001).

2.4. *Other Natural Language Processing Components*

The language analysis in the prototype system is limited. We currently only apply techniques in noun phrase parsing, and simple language generation. The components in the dotted boxes in Fig. 1 are placeholders and will be addressed in the next version of the system.

The noun phrase parser is used to process grammatical and semantic information of interest from the user natural language input. This shallow parser extracts the head of the noun phrase from its modifiers and identifies key concepts of interest that are present in the user's query. For example, if the user is looking for a "thinkpad with external mouse", the parser identifies the "laptop" and "mouse" concepts and marks the "laptop" as the headword, which represents the item the user is interested in. On the other hand, if the user enters "mouse for laptops," the parser identifies the same concepts, but marks "mouse" as the headword. Parsing, as opposed to pure keyword matching, improves performance by distinguishing between head and attributes and identifying key concepts rather than keywords.

The natural language generation is applied when the system recommends products to the user. In the prototype system, when a product is recommended, an explanation of the recommendation will be generated automatically. It integrates the concepts detected in the user's utterance from the history of interactions.

2.5. *Screen Shots and a Walkthrough*

Several screen shots in Fig. 2 show an example of interactions. Screen 1 provides a text field for the user's NL query. A carrier phrase, "I'm looking for," is given in the anticipation that users will follow with descriptive noun phrases. In this example, the user types in the query ("a notebook for my consulting business.") The PM applies the Noun Phrase Parser and detects the "thinkpad" and "business-use" concepts. The DM receives two concepts and checks them with the Knowledge Base for business rules. Some rules are partially matched. The DM sends these potential rules to the AM. The AM shows a list of interesting items corresponding to the potential rules, together with brief descriptions for the user to browse (as in Screen 2). In addition, the DM compares ranks and weights of the potential rules and prompts a question ("Please describe your final constraints") for more information. The user can browse the interesting items as he/she wishes or can choose to answer the question to narrow down the search space. In Screen 2, the user types in "not important, but the performance is essential". Based on this input, the PM discards the "affordability" concept and detects the "high-performance" concept. The DM matches the previously active rules and sends a further narrowed set of potential rules to the AM. The AM will retrieve a list of high performance models and show them to the user. Once again, by comparing ranks and weights of the rules, the DM prompts another question ("are you looking for something that is top of the line?") (as in Screen 3). The answer of "yes, absolutely" triggers the "best-performance" concept and has a complete match with a business rule for Thinkpad 770 (in Screen 4). Based on the interaction history, a paragraph (summary) is generated to explain to the user why that product is recommended. The user can follow the "here" link or the picture icon for the additional information about Thinkpad 770.

3. Comparative Evaluation

We conducted a user study to evaluate the natural language dialog-based prototype system in comparison with a fully developed (by an independent organization) menu-driven system. For the user testing, we addressed the following questions: Can natural language based navigation be more efficient (number of clicks, time spent searching, etc.) and easier to use than menu-driven navigation? By how much? What are users'

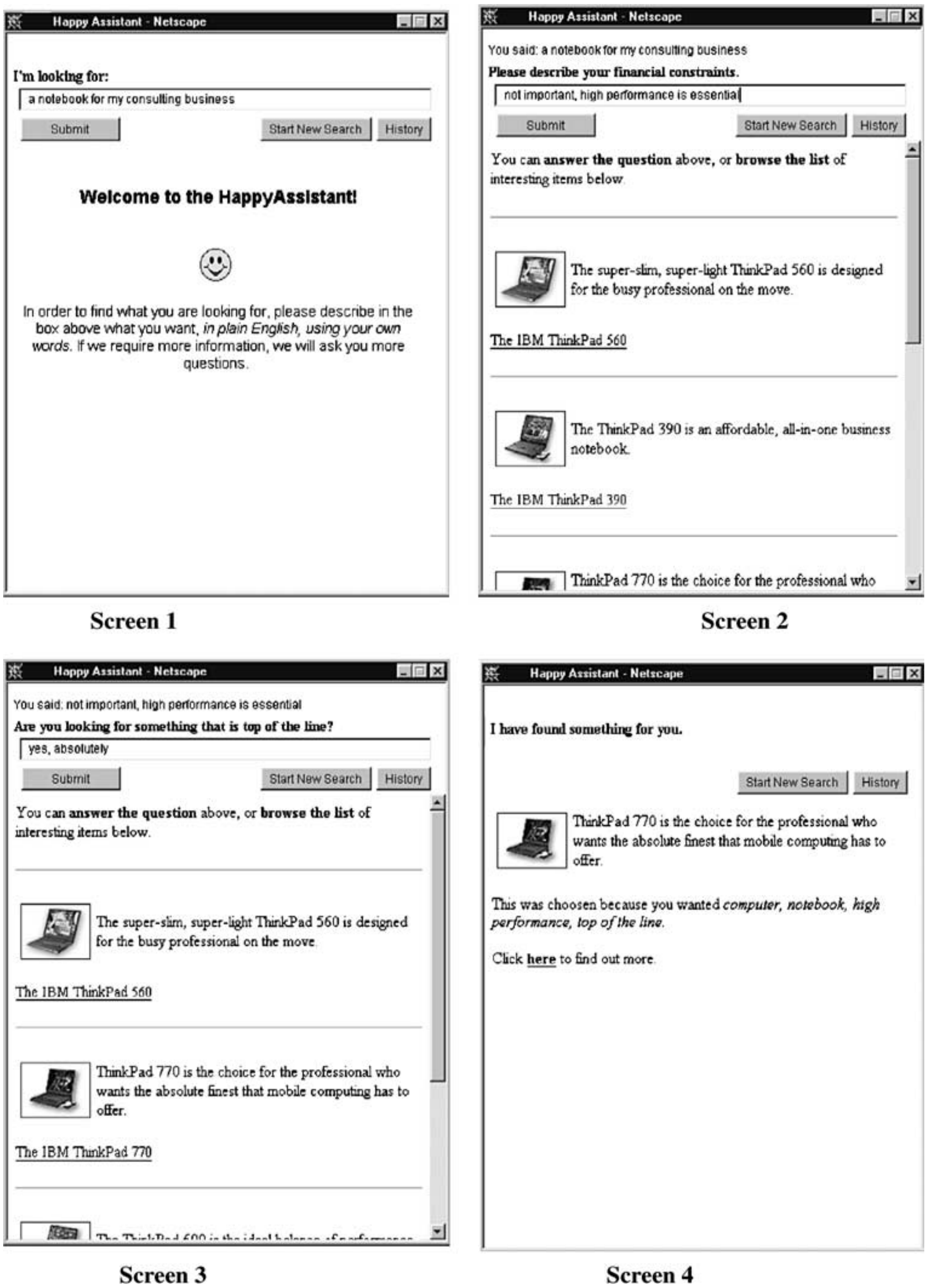

Figure 2. Examples of interactions.

responses toward natural language-based navigation as opposed to menu-driven navigation? How do users with different online experiences react to the natural language dialog-based navigation?

3.1. Menu-Driven System

Menu-driven systems are commonly used navigation devices on business oriented Web sites. Typically, menu-driven systems offer a limited number of options from which to choose by displaying radio buttons, choice boxes, or pull-down menus. The system used in the user testing is based on a question/answer paradigm. The system prompts the user with a question and a list of answers. When the user chooses one answer, the system will provide another question with a list of answers. The process continues until the system exhausts all predefined questions/answers and reaches the final recommendation. An example of this kind of structure can be found in Fig. 3.

3.2. Testing Background

An independent testing agency recruited a total of 17 people. They tested the natural language dialog-based system and the menu-driven system. A screener was used to recruit the participants. Among those participants, four of them considered themselves to have advanced computer skills, eight felt they had an intermediate level of proficiency and five admitted to limited experiences with the internet.

A testing room was set up with a division to allow for the respondent and the moderator to work at one monitor while we were positioned behind the divider at another monitor. The systems were linked in such a way that we could manipulate the Happy Assistant prototype if necessary and observe the testing as it occurred without interfering with the interview. Such manipulation was intended only for some fatal errors like infinite loops or unexpected exceptions, and was rarely used. In the testing, it turned out that less than 5% of the interactions required intervention.

Each interview began with an introduction by the moderator explaining the purpose of the interview. It was explained that subjects would be using two prototype web sites. In addition, they were informed of the moderator's independent and objective position and encouraged to be open with their opinions of the prototype. After the introduction, the participants were given various scenarios. These scenarios were designed to let them experience critical parts and navigation of each web site in order to form an opinion of the tool's concept. They were then asked to rank the scenarios on a 1 to 10 scale (where 10 is easy) with regard to the ease of navigation and the series of events leading up to the result. The moderator probed the participant throughout the scenario, and the participants were asked for their overall reaction to the concept of each prototype upon completion. In total, six scenarios were used for the testing. For each scenario, there were two similar versions presented: one for the natural language system and the other for the menu-driven system. Each participant was randomly assigned three scenarios.[1]

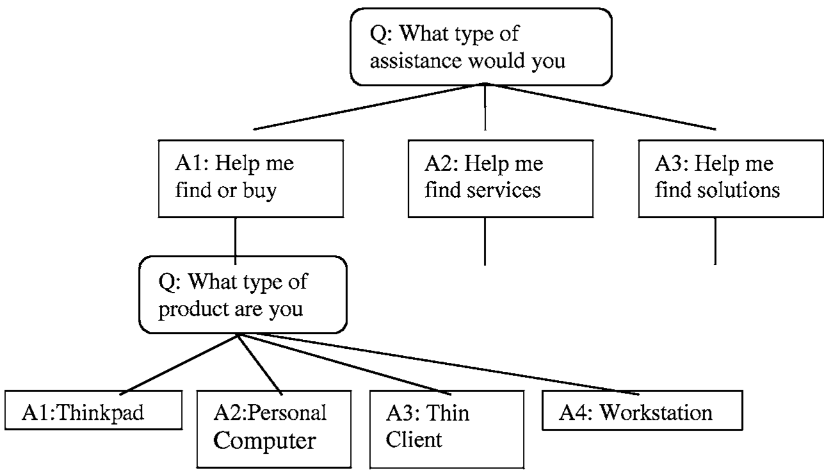

Figure 3. Example of menu navigation structure.

3.3. Observations and Results

Our first evaluation compared NL dialog-based navigation with menu-driven navigation in terms of the number of clicks and the amount of time required to accomplish a certain task. A click was counted when the user clicked on something (e.g., a submit button, a radio button, a link) to take action. For example, in the HappyAssistant, a click could be the click of the "submit" button after the user typed in natural language input either to specify his/her needs or answer questions. In the menu-driven system, a click could be the click on a particular choice of the answer to a question. The results showed that, in the HappyAssistant, the number of clicks was significantly reduced by 63.5% (indicated by the *T*-test, $P < 0.0005$) and the amount of time spent was significantly reduced by 33.3% (indicated by the *T*-test, $P < 0.025$), relative to the menu-driven system. The HappyAssistant requires less time and fewer user movements (mouse clicks) overall than the menu-driven system. With the exception of a few, the majority of the respondents perceived the HappyAssistant to be a faster, more efficient process. Moreover, many appreciated the minimal number of screens required to achieve the desired product. The greatest amount of time taken by respondents with the HappyAssistant was used to formulate the words required to express their needs of a particular scenario in the study. It is expected that this time would be greatly diminished when a user is describing his/her own needs and not those of a study. The comparison can be found in Fig. 4. The horizontal axis is the scenario number, and the vertical axis represents the metrics of interest.

After each task, we asked participants to rate the ease of use of the two systems. The average rating of subjects with limited internet experience was 9.4 for the natural language-based system and 6.3 for the menu-driven system. The average rating of users with intermediate experience was 8.5 for the natural language system and 8.1 for the menu-driven navigation. The average rating of users with advanced experience was 8.3 for the natural language system and 8.9 for the menu-driven navigation. Figure 5 shows the average ratings for each scenario by three different groups. Because the scenarios were randomly assigned, it turned out that no one with limited experience tested scenario 5 and no one with advanced experience tested scenarios 1 and 6. This result showed that the less experienced users preferred the NL-enabled navigation much more than did the experienced users.

The vast majority of user queries (85%) were relatively short, and consisted of noun phrases with attached prepositional phrases or lists of keywords. Despite the moderator's assurance that the users could type "anything they wanted," complete sentences were seldom observed. Analysis of the user input revealed that the average length of a user query was 5.31 words long (with a standard deviation of 2.62). Some sample inputs include: "moderately priced laptop", "computer with internet access + games", and "a high-speed computer."

Overall, respondents preferred the NL dialog-based navigation (HappyAssistant) to the menu-driven navigation by two to one (2:1). Respondents thought the HappyAssistant was extremely easy to use, and they were comfortable and confident with the resulting information it provided. Users liked the fact that they could express their needs in their "own jargon" instead of the foreign "computer jargon". There was also the perception that with the HappyAssistant model, the

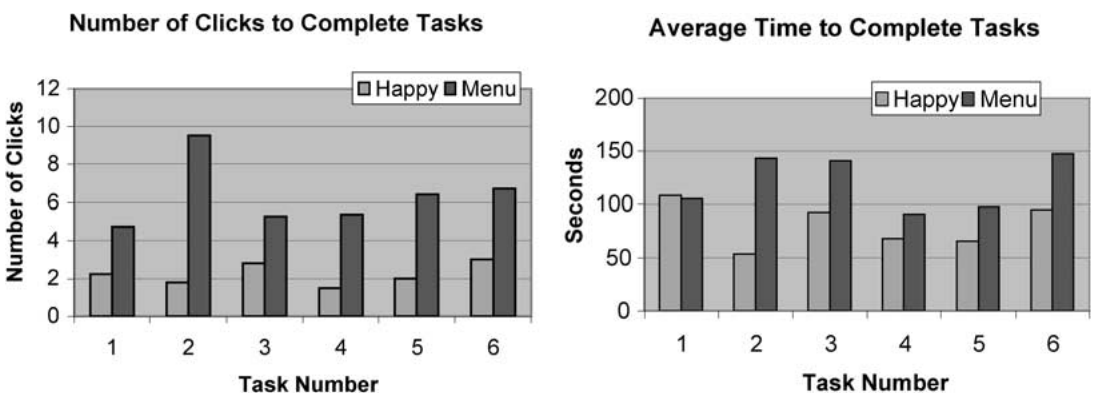

Figure 4. Number of clicks and average time spent to accomplish each task.

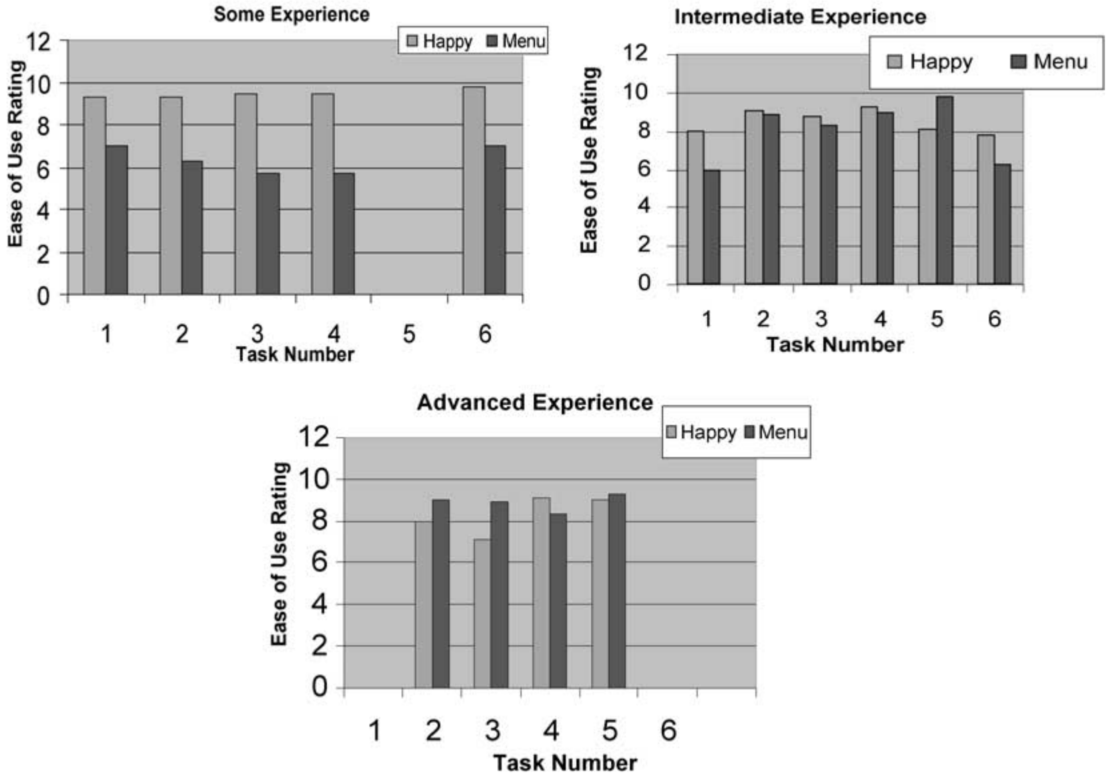

Figure 5. Ratings from users with different level of experience.

computer did all the work for them instead of them doing all the work for the computer (as in the menu-driven model).

Respondents were extremely pleased with the personal search instead of the generalization of the individual's needs. Many commented positively on the pronouns used by the site such as "***I** have found something for **you***". The personalized care makes them "feel like an individual" and not like "70,000 other people". Moreover, many respondents preferred to type what they were looking for and not to answer generic questions that did not apply to their needs.

Despite the fact that most users preferred the NL dialog-based navigation, there was great evidence of the utility of menu-driven searches. There were definitely users who liked the ability to select options from a menu, specifying that the multiple-choice method was easy. There were also users who liked having questions asked of them. Typically, such users were either not comfortable with their typing ability or unable to express what they were looking for without additional information.

3.4. *Discussion*

In this study, we have found that natural language dialog shortens the interaction time and provides a more natural interface that users prefer, especially less experienced ones. The current internet keyword search engines have created a "search culture" which is widely accepted by most internet users. As a result, many users are accustomed to typing keywords or simple phrases. Analysis of the user queries reveals the brevity and relative linguistic simplicity of their input; hence, shallow parsing techniques seem adequate to extract the necessary meaning from the user input. Therefore, in such contexts, sophisticated dialog management is more important than the ability to handle complex natural language sentences.

Furthermore, there are additional practical issues that affect the application of NL-based dialog systems. Many users are not very confident of their typing abilities, and hence tend to give shorter and more general descriptions of their needs. Also, spelling is an important factor and spell-check would be a great feature in the typing-based interface. The addition of a speech interface is desirable.

We have also learned that in order to improve the functionality of an e-business site, the natural language dialog navigation and the menu-driven navigation should be combined to meet users' different needs. While the menu-driven approach can provide choices for the user to browse or learn some additional information, the natural language dialog provides the efficiency, flexibility and natural touch to the users' online experience.

Moreover, in designing NL dialog-based navigation, one important issue is to show users that the system does understand their requests before giving any recommendation or relevant information. Users remarked in our study that they appreciated the recommended results because the system offered some explanation. This feature allows the user to "trust the system." Good navigation aids can be provided to summarize the user's requests by paraphrasing them using context history, or by engaging in meaningful conversations with the user. Our study found that robust natural dialogue had a very big influence on user satisfaction—almost all of the respondents appreciated the additional questions prompted by their input and the summary of their previous queries.

4. Conclusions

This paper describes a prototype system that provides natural language dialog capabilities to help users access e-commerce sites to find relevant information about products and services. The prototype system leverages technologies in natural language processing and human-computer interaction to create a faster and more intuitive way of interacting with websites, especially for less experienced users. The result of a comparative study shows that users prefer the natural language-enabled navigation, two to one over the menu-driven navigation. In addition, the study confirmed the efficiency of using natural language dialog in terms of the number of clicks and the amount of time required to obtain the relevant information. Compared to the menu-driven system, the average number of clicks used in the natural language system was reduced by 63.2%, and the average time was reduced by 33.3%.

The work presented in this paper is a proof-of-concept study. Although the prototype system has employed only basic techniques in natural language processing and human-computer interaction, the results learned from the user testing are significant. By comparing this simple prototype system with a fully deployed menu system, we have learned that users, especially novice users, strongly prefer the natural language dialog-based system. We have also learned that in an e-commerce environment sophistication in dialog management is more important than the ability to handle complex natural language sentences. Furthermore, to provide easy access to information on e-commerce sites, natural language dialog-based navigation and menu-driven navigation should be intelligently combined to satisfy user's different needs.

Recently, we have completed development of a new generation of the system that includes tremendous enhancements in language processing, dialog management and data management (Chai et al., 2001). We believe that natural language conversational interfaces offer powerful personalized alternatives to traditional menu-driven or search-based interfaces to websites.

Note

1. An example of a scenario is: one version: "Your daughter's birthday is coming soon. Currently she is a high school student and you would like to get her a computer for birthday. You would also like to have internet access at home and occasionally play some computer games with your kids," and the other version: "You have just moved into a new house and would like to get a computer for your home. You are interested in internet access to check email, browse websites, and trade online from home. Occasionally, you would also need to write letters."

References

Aggarwal, C., Wolf, J., and Yu, P. (1998). A framework for the optimizing of WWW advertising. In W. Lamersdorf and M. Merz (Eds.), *Trends in Distributed Systems for Electronic Commerce. LNCS 1402*, Berlin: Springer.

Androutsopoulos, I. and Ritchie, G.D. (1995). Natural language interfaces to databases—An introduction. *Natural Language Engineering 1*(1):29–81.

Bennacef, S., Devillers, L., Rosset, S., and Lamel, L. (1996). Dialog in the RAILTEL telephone-based system. *Proceedings of International Conference on Spoken Language Processing, ICSLP'96*, Philadelphia, USA, vol. 1, pp. 550–553.

Bray, T., Paoli, J., and Sperberg-McQueen, C.M. (1998). Extensible markup language (XML) 1.0. (Technical Report). *http://www.w3.org/TR/REC-xml-19980210*, World Wide Web Consortium Recommendation.

Carpenter, B. and Chu-Carroll, J. (1998). Natural language call routing: A robust, self-organizing approach. *Proceedings of the Fifth International Conference on Spoken Language Processing*, pp. 2059–2062.

Chai, J., Horvath, V., Nicolov, N., Stys-Budzikowska, M., Kambhatla, N., and Zadrozny, W. (2001). Natural language sales assistant—A web-based dialog system for online sales. *Proceedings of Thirteenth Annual Conference on Innovative Applications of Artificial Intelligence*, Seattle, WA, August, 2001, to appear.

Chu-Carroll, J. and Carpenter, B. (1998). Dialog management in vector-based call routing. *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics*, pp. 256–262.

Ferguson, G., Allen, J., and Miller, B. (1996). TRAINS-95: Towards a mixed-initiative planning assistant. *Proceedings of the Third Conference on Artificial Intelligence Planning Systems, AIPS-96*, pp. 70–77.

Huberman, B.A., Pirolli, P.L.T., Pitkow, J.E., and Lukose, R.M. (1998). Strong regularities in world wide web surfing. *Science*, *280*: 95–97.

Muller, J. and Pischel, M. (1999). Doing business in the information marketplace. *Proceedings of the 1999 International Conference on Autonomous Agents*, Seattle, USA, pp. 139–146.

Radev, D., Kambhatla, N., Ye, Y., Wolf, C., and Zadrozny, W. (1999). DSML: A proposal for XML standards for messaging between components of a natural language dialog system. *Proceedings of the AISB'99 (Artificial Intelligence and Simulation of Behavior) Workshop on Reference Architecture and Data Standards for NLP*, Edinburgh, England.

Saito, M. and Ohmura, K. (1998). A cognitive model for searching for ill-defined targets on the web—The relationship between search strategies and user satisfaction. *Proceedings of 21st International Conference on Research and Development in Information Retrieval*, Australia, pp. 155–163.

Seneff, S., Goddeau, D., Pao, C., and Polifroni, J. (1996). Multimodal discourse modeling in a multi-user multi-domain environment. *Proceedings of International Conference on Spoken Language Processing, ICSLP'96*, Philadelphia, USA, pp. 193–195.

Walker, M., Fromer, J., and Narayanan, S. (1998). Learning optimal dialogue strategies: A case study of a spoken dialogue agent for email. *Proceedings of 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, Montreal, Canada, pp. 1345–1351.

Zadrozny, W., Wolf, C., Kambhatla, N., and Ye, Y. (1998). Conversation machines for transaction processing. *Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI) and Tenth Conference on Innovative Applications of Artificial Intelligence Conference (IAAI)*, Madison, Wisconsin, USA, pp. 1160–1166.