From: AAAI Technical Report SS-03-06. Compilation copyright © 2003, AAAI (www.aaai.org). All rights reserved.

Fully Corpus-Based Natural Language Dialogue System

Nobuo Inui, Takuya Koiso, Junpei Nakamura and Yoshiyuki Kotani

Department of Computer, Information and Communication Sciences, Tokyo University of Agriculture and Technology
2-24-16 Nakacho Koganei Tokyo, 184-8588, Japan
E-Mail: {nobu, tatsu, naka-jun, kotani}@fairy.ei.tuat.ac.jp

Abstract

We describe a corpus-based approach of natural language dialogue system. The characteristic is that all the system's behaviors, like processing and understanding dialogues and generating responses, depend on corpora. As a result, the system can handle any language and any topic. This paper aims to explain the whole architecture and individual technology used in our project.

Corpus-based Natural Language Dialogue System

Natural language dialogue between human and computer is an efficient way of communication. Natural language interfaces (NLIs) are successful systems for help systems, secretariat systems and so on. The key of these systems is to handle restricted and well-known domains. In these domains, the designers can express the rules of understanding user's requests and generating system's responses. In contrast with such domain-specific NLIs, we have been designing a general-purpose NLI system. The aim of our research is to explore friendly dialogues between human and computer without giving human-made rules to the system. The goal of our research is to understand what are the factors of natural dialogues by corpus-based approaches.

To explore the general-purpose natural language dialogue (NLD), our method is to use linguistic corpora. The corpora are sets of sentences. As the counter method, we can use general rules like ELIZA [Weizenbaum 1966]-type dialogue system. But this kind of system merely responds an informative answer to users. Against the general-rule-based system, our corpus-based system can answer user's questions within the description of corpora. Since almost articles of news, novel, homepages and so on are available electrically, the corpus-based system accumulates knowledge beyond human ability. By the use of various corpora, our system is expected to be a good adviser.

We propose a method of using only corpus in this paper. The merits of our method are:

(1) Language-independent
(2) Topic-independent.

We adopt corpus-based methods like stochastic model, N-gram model, keyword matching, and structural matching. No linguistic and conceptual knowledge are used for our system. This makes it possible to use any language corpora. In addition, the result of analysis depends on the corpora. This means that if a baseball corpus is used, the system answer your question within terms of baseball. Using our approaches, the only thing that system providers should do is to collect the linguistic corpora for dialogues to go well.

This paper describes the architecture of our fully corpus-based NLD system in the next section. From the next section, the individual technology of designing each component, analyzing inputted sentence, searching resemble cases in a database and generating outputted sentence are described.

Outline of Corpus-Based Natural Language Dialogue System

Fig. 1 illustrates the whole architecture of our system.

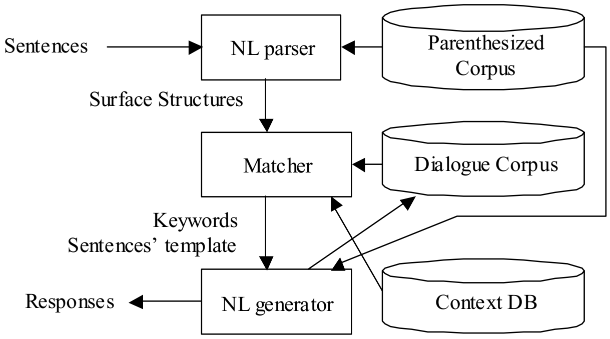

Fig.1 Architecture of Fully Corpus-Based NLD System

The purpose of the Natural Language (NL) parser is to analyze inputted sentences. Since, in our approaches, we consider that the dependency structure is only needed for matching the dialogue corpus, the NL parser generates surface structures of sentences. To analyze sentences, the NL parser uses parenthesized corpus [EDR 1996] that includes parenthesized sentences.

The Matcher searches the most resemble dialogue from the Dialogue Corpus. The Dialogue Corpus contains the series of dialogues. The Context DB holds dialogue acts (DA) that are the intention of sentences like greeting, question, explain and so on. The Matcher is going to find the most resemble dialogue of the current flow of dialogue using the Dialogue Corpus and the Context DB. The

Matcher generates keywords along with the current dialogue and sentences' templates to make responses.

The NL generator generates the system's responses. The role of the NL generator is to determine the exchanges of keywords and words in sentences' templates. The system designer gives the exchange strategy because it decides the character of the system. The structures and the word orders of responses are determined using the parenthesized corpus and replacing words simply. The inputted sentences and the responses are stored to the Dialogue Corpus to re-use for the future dialogue.

Natural Language Parser

Since it is possible for a user to input ungrammatical sentences, the NL parser is required to parse any sentences robustly. For this reason, we use the N-gram-based shallow parser [Inui 2002a][Inui 2001] shown in Fig. 2. A large parenthesized corpus is easily available.

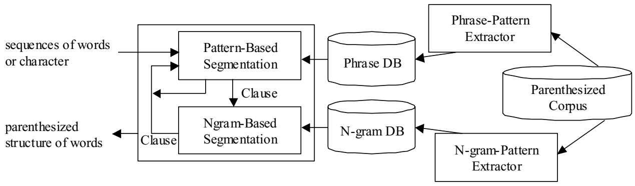

Fig.2 The Architecture of NL Parser

The characteristic of this parser is that no ready-made syntactic rules like context-free based rules [Charniak 1997] are required. The Phrase-Pattern Extractor generates the Phrase DB and N-gram-Pattern Extractor generates the N-gram DB from the parenthesized corpus. To parse sentences robustly, the N-gram-Based Segmentation is designed. In both of segmentation modules, a clause boundary marker (@, described later) is introduced as a special word. The Pattern-Based Segmentation module simply divides a sequence of words (sometimes characters) into a sequence of clauses, based on the frequency of clause (for example, ((I) (went to Palo Alto))). If the parser fails to find the pattern, the sequence of words is handed to the N-gram-Based Segmentation module.

The N-gram-Based Segmentation module tries to parse a sequence of words by N-gram DB which contains N-gram rules like, for example in case of bi-gram, 'I @', '@ went', 'went to' and so on. The N-gram-Based Segmentation module uses the conditional probability to forecast positions of clause boundary markers so as to maximize the expression (1).

$$(1)\,P(w_1 \cdots w_n) = P(w_1)P(w_2 \mid w_1)\cdots P(w_n \mid w_1 \cdots w_{n-1})$$

$$\approx P(w_1)P(w_2 \mid w_1)\cdots P(w_n \mid w_{n-k+1} \cdots w_{n-1})$$

$$P(w_{j-k+1} \cdots w_n) = \frac{freq(w_{j-k+1} \cdots w_n)}{total\ frequency}$$

We use the linear interpolation expression of the expression (1) for the robust parsing. Finally, surface structures like ((I)((went) ((to)(Palo) (Alto))))) are generated in the NL parser by only using the parenthesized corpus.

Matcher

The aim of the Matcher is to find the most resemble dialogue for the current dialogue from the Dialogue Corpus. We designed two kinds of implementation for the Matcher, key words matching with dialogue acts and structural matching. The architecture of the Matcher is shown in Fig. 3.

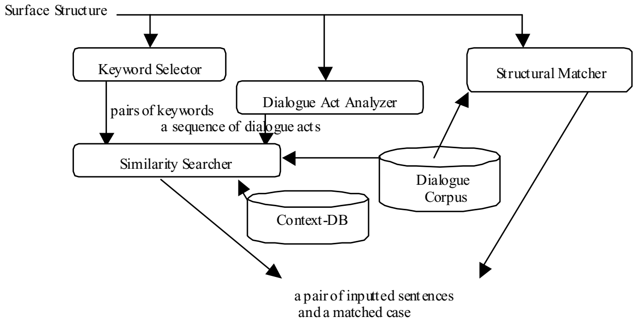

Fig.3 The architecture of the Matcher

In the keyword-based matcher, nouns and verbs are considered to express the essence of sentences. But it is difficult to find thematic noun and verb without knowledge. To cope with this issue, the Similarity Searcher determines the best noun and verb among all combinations of nouns and verbs from the Dialogue Corpus (DC). In addition, to express the flow of dialogue, we use a sequence of dialogue acts [Reithinger 1997]. For example, YES/NO-type sentences often appear after QUSTION-type sentences. Such knowledge is collected from DC. We use the stochastic estimation of dialogue acts [Inui 2001b] in the Dialogue Act Analyzer. Finally, the Similarity Searcher finds the most resemble dialogue (i.e. including the same keywords and the dialogue acts) from DC. For example, the Similarity Searcher searches sentences in DC with "Palo Alto" and "went" as keywords and "explain fact" as a dialogue act.

The Structural Matcher [Koiso 2002] is another way to find the most resemble case. In this case, the similarity is calculated using the structural distance between two sentences. Such structural information is considered to include keywords and dialogue acts. In our current implementation, a user selects a matcher before he uses our system.

In both cases, the next sentences of the most resemble sentences becomes candidates of the responses by the system. For example, consider a sequence of dialogue, A, B, C, D and the most resemble sentence of the current inputted sentence is B. A sentence C becomes a candidate of sentence template of the response. C would be used for the next response.

It is also possible to find an appropriate dialogue by looking up the n-past sentences. For example, the Matcher tries to make a correspondence with a sequence of sentences A, B and C. If C is not appropriate for the current dialogue, we can find a sentence D as a candidate of the next utterance.

It is possible to include picture character like ;-<, :-) and so on. By using picture characters, a user feels friendly to use the dialogue system [Nakamura 2002]. We consider that this kind of response is important to continue dialogues for human.

Natural Language Generator

The aim of the Natural Language generator is to generate responses to a user. Fig. 4 illustrates the architecture.

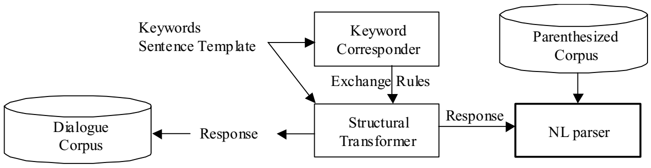

Fig. 4 The architecture of the NL generator

The Keyword Corresponder makes the correspondence between keywords in inputted sentences and the word in sentence templates. For example, when "I went to Palo Alto" is matched to "He went to Africa", the keyword corresponder makes rules like exchanging 'He' to 'I' and exchanging 'Africa' to 'Palo Alto'. This is made by the structural positions of words in sentences.

The Structural Transformer generates several candidates of responses using exchange rules. The NL parser described in the section 3 determines which sentence is preferable. The NL parser outputs the structure and the occurrence probability of a sentence. The Structural Transformer selects the preferable response with the maximum occurrence probability. This guarantees the grammatical correctness of the response.

Finally, the response is given to user as the system's utterance and the current dialogue is stored to the Dialogue Corpus. The current dialogue would be re-used for the future dialogue. In this manner, the system can grow the Dialogue Corpus automatically.

Summary of Example Dialogue

We demonstrate an example of dialogue explained above.

User: I went to Palo Alto
NL parser: ((I) (went (to (Palo Alto))))
I: noun, went: verb: to: prep., Palo Alto:noun
Matcher: He went to Africa, *How did he go there?*
NL generator: *How did I go there?*
System: How did I go there?

In this case, 'I' should be re-rewritten to 'you' in the system's utterance. Our method cannot do this, because none of knowledge about person is acquired. But we can improve our system's response by collecting enough dialogues.

Conclusion

We described our fully corpus-based natural language dialogue system in this paper. We are carrying the evaluation of our system out. The merit of our method is that we can tune the system behavior to our favorite one by changing corpora. We are interested in connecting a voice-recognition system and a voice-generating system to our dialogue system. We consider our method as a tool for exploring the naturalness and the friendliness of dialogue for human.

Acknowledgement

This research was supported by the grant in aid No. 14580409, Japanese Ministry of Education, Culture, Sports, Science and Technology

References

[Charniak 1997] Charniak E. 1997. *Statistical Parsing with a Context-free Grammar and Word Statistics*. Proc. AAAI 97. pp.598-603.

[EDR 1996] EDR. 1996. *EDR Electric Dictionary Manual Ver. 1.5*.

[Koiso 2002] Koiso T., Ikeda T., Inui N., and Kotani Y. 2002. *A dialog system which chooses a response using similarity between a surface case rule patterns*. IPSJ conference. 1M-03.

[Inui 2001a] Inui N. and Kotani Y. 2001. *Robust N-gram Based Syntactic Analysis Using Segmentation Words*. 15th PACLIC. pp.333-343.

[Inui 2001b] Inui N., Ebe T., Indurkhya B., and Kotani Y. 2001. *A Case-Based Natural Language Dialogue System Using Dialogue Acts*. IEEE International Conference on Systems, Man and Cybernetics. pp.193-198.

[Inui 2002] Inui N. and Kotani Y. 2002. *Using Patterns for Syntactic Parsing*. IASTED International Conference Artificial Intelligence and Appications. pp.522-527.

[Nakamura 2002] Nakamura J., Ebe T., Ikeda T. Inui N. and Kotani Y. 2002. *A method of outputting FACEMARK suitable for the response sentence of natural dialogue system using emotions model and DIALOGUE ACT*. IPSJ Conference. 1M-04.

[Reithinger 1997] Reithinger N. and Klesen M. 1997. *Dialogue Act Classification Using Language Models*. EuroSpeech-97. pp.2235-2238.

[Weizenbaum 1966] Weizenbaum J. 1966. *ELIZA- A Computer Program for the Study of Natural Language Communications between Men and Machines*. CACM. Vol. 9. pp.3-45.