Design of a Semantic Parser with Support to Ellipsis Resolution
in a Chinese Spoken Language Dialogue System

Yi Su, Fang Zheng, and Yinfei Huang

Center of Speech Technology, State Key Laboratory of Intelligent Technology and Systems, Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China
{suy, fzheng, hifi}@sp.cs.tsinghua.edu.cn, http://sp.cs.tsinghua.edu.cn

Abstract

In this paper, a semantic parser with support to ellipsis resolution in a Chinese spoken language dialogue system is proposed. The grammar and parsing strategy of this parser is designed to address the characteristics of spoken language and to support the ellipsis resolution. Namely, it parses the user utterance with a domain-specific semantic grammar based on a template-filling approach. Syntactic constraints extracted by a Generalized LR parser are also used in the parsing process. With a paradigm of two-state bottom-up parsing and a scoring scheme, the ellipsis resolution module is integrated into the parser seamlessly. The parsing result is represented by a linked structure of semantic frames, which is convenient to both the parser and its successive components of the dialogue system.

1. Introduction

Much work has been done to understand the spoken language in a dialogue system. Conventional parsing techniques in the natural language processing are proved to be insufficient since they are developed for the written language. To deal with the spoken language, which is often ungrammatical, sometimes ill formed, and degraded by recognition errors, there are two major categories of approaches [1]. In syntax-driven formulations, a complete syntactic resolution is performed which attempts to account for all words in an utterance [2], while semantic-driven approaches tend to obtain their understanding by spotting key words and phrases in the utterance [3]. By focusing on the information that the user is willing to convey instead of the rigid syntactic analysis of his/her utterances, this method achieves a satisfying result even on some syntactically ill formed yet semantically sound sentences, which are common in the spoken language. Additionally, a partial parse can be valuable to recover the recognition errors of the front end to some extent [1].

The ellipsis is another aspect in which the spoken language differs from the written language, which is syntactically integrated in most of the cases. In fact, utterances may become incomprehensible without regard to their context in the spoken language. Where to perform the ellipsis resolution is something arguable. Some systems tend to deal with the ellipsis resolution as a post-processing of the semantic parsing in a rather simple manner [3] while others would like to do this job in the dialogue manager [4]. Both are applicable.

In this paper proposed is a semantic parser with support to ellipsis resolution in a Chinese spoken language dialogue system. With this parser, we attempt to parse flexible spoken language using a semantic grammar in a bottom-up manner. Syntax constraints are also used for disambiguating. And with a paradigm of two-state parsing and a useful scoring scheme, it provides a strong support to the ellipsis resolution.

The architecture of our spoken language dialogue system is briefly introduced at first in the following section. Then the design of our language understanding component, the parser, is described in detail in Section 3. Finally, the conclusions are drawn in Section 4.

2. Architecture of the spoken language system

Our spoken language dialogue system is a mixed initiative Chinese campus navigation system for Tsinghua University, named EasyNav, consisting of a speech recognizer, a linguistic processor, a dialogue manager and a response generator (See Figure 1.). Early development of our system can be found in [5]. In contrast to other systems, ours focuses on the linguistic processing module instead of the dialogue management module. Because in our task the user is usually more active than the computer to get information from the opposite side, i.e. the user initiates the dialogue in most of the time and speaks very freely. Therefore understanding what the user says emerges to be the most crucial factor to the system.

Figure 1. Architecture of the spoken language dialogue system: EasyNav
After a user utterance is recognized, the first candidate sentence will be passed to the linguistic processor for the syntactic parsing, the semantic parsing and the ellipsis resolution. The understanding of the utterance with regard to the ellipsis resolution, represented by a set of nested frame-slot structures, is passed to the dialogue management component then. The manager will simply invoke the database query module to get the desired information or will decide whether a clarification of the user’s intention is needed. Finally the system response is generated accordingly and transformed into a waveform through a Text-to-Speech engine so as to accomplish a turn of human-computer interaction.

3. Design of the language understanding component

3.1. Overview

The block diagram of our language-understanding component is depicted in Figure 2. It is significantly different from its former version [5]. This parser is essentially semantic-driven while its ancestor is actually a syntactic-driven one, in other words, it will always try to carry out a full syntactic parsing whenever a user utterance is available to the system.

The recognition result from the recognizer, represented as a sequence of Chinese characters, is first segmented into words because there is no explicit boundary between Chinese words. The word segmentation is not the focus of this paper and thus will not be discussed deeply. In fact, it is still an open issue in Chinese information processing. Though the word boundaries can be obtained from the recognizer alternatively, adopted in our system is an N-gram based word decoding approach which is similar to our previous proposal in speech recognition [6].

The word sequence is then transferred to a Part-of-Speech (POS) tagger and a semantic labeler respectively. The POS tagger attempts to transform it into a sequence of POS tags. Like the Chinese word segmentation, the POS tagging is also something worthy of efforts but of no interests in this paper. We simply use the word category N-gram probabilities and the word probabilities given their categories to calculate the probabilities of the POS sequence candidates. The candidate with the maximal likelihood will be chosen as the result. At the same time, the semantic labeler converts the word sequence into a sequence of semantic labels/categories with a semantic lexicon. The semantic lexicon is a list of words with the categories to which they belong. For instance, “二校门”(The Second Gate), a famous sight in Tsinghua University, is labeled as a location. And “什么地方”(what location), “什么位置”(what position), “哪儿”(where), “哪”(where), “哪里”(where), “哪个地方”(which location), and “哪个位置”(which position) all belong to the category “where”. Thus only the words with concrete meanings are retained for the following analysis. Since the dialogue system is not a universal one, the lexicon is domain-specific.

A Generalized LR (GLR) parser is then used to parse the POS sequence [7]. The GLR algorithm is proposed as an extension of the traditional LR algorithm to parse a more general context free grammar than the LR grammar. Since a general grammar that can cover almost all Chinese sentences is too large and it is not necessary for the spoken language, we reduce the grammar to fit our specific domain and the characteristics of the spoken language. In result it provides us with much more instructions on understanding the user’s utterances than a general one does.

Finally, the semantic label sequence is passed to and parsed by the semantic parser with a manually crafted domain-specific semantic grammar and the syntactic constraints extracted by the GLR parser. The parsing result is represented by a linked semantic frame-slot structure, which is convenient for both the database query module and the response generator. When needed, the semantic parser may refer to the ellipsis resolution module to solve the ellipsis problem. More details will be discussed in the following sections.

3.2. Semantic grammar

The parser uses a set of semantic context free rules as the grammar, which defines the way that different semantic labels are assembled to form a new one under certain syntactic constraints along with actions of filling the slots of a semantic frame. An example is given in Figure 3.

This rule indicates that an attribute can be made of a location and a nearby providing that the location and the nearby compose a prepositional phrase (pp). The italic words stand for semantic labels, which are actually semantic categories of words or phrases. The dot (.) following the location indicates that it can be omitted. The “set_area()” is a query function implemented in the database query module.
The grammar takes the semantic labels defined in the semantic lexicon mentioned above as its terminal tokens and some other pre-defined semantic categories as its non-terminal ones. Each category or label is a semantic independent unit, in other words, each has a substantial meaning corresponding to one or more phrases. Some requirements in syntax, such as “<pp>” in the example above, are attached to the rule. The syntactic restraints are applied in the phrase level for two reasons. First, even for some syntactically ill-formed utterances, their phrases are often well formed. It seems that the spoken language is “phrase-based” while the written language is “sentence-based”. Second, applying the syntactic restraints at the phrase level is more flexible than applying them at the whole sentence level. Actually in the earlier version of our system, the syntactic restraints are imposed to the whole sentence [5]. While getting simplicity to a certain extent this strategy loses much flexibility.

3.3. Semantic frame

The output of the semantic parsing must be a certain representation of semantics. A linked frame-slot structure with domain specific slots is used, namely, the meaning of an utterance is described by a set of semantic frames, among which a slot of a frame may refer to a slot of another frame. For instance, the linked frames for “九号楼在哪儿” (“where is the No. 9 Building”) is depicted in Figure 4.

As you may see, the first frame named “query_for_location” acts as the “root” frame, with a slot “reply” indicating the slot for which the user asks and thus whose value should be queried from the database. The database query procedure can be described as an evaluation of the “reply” slot. That is, a set of basic functions, such as Query_location (location_name), is implemented inside the database query module first. Then they are called with a proper sequence and combination according to the user’s intention, delivered by the linked semantic frames. In fact a simple traverse of the frames will give the right order of evaluation. With the linked structure of the semantic frames, a complicated combination of the basic query functions becomes possible. Thus the user can express her/his purpose directly in one utterance instead of paraphrase it into several simple sentences to be accustomed to the dialogue system. In other words the user can speak more naturally with ease.

While one reason for using linked semantic frames instead of simply one frame is the flexibility, another reason is that it can be easily constructed during the parsing process accompanying the construction of the parsing tree. In fact, the linked frames themselves are not only the representation of the parsing result but also that of the parsing process. One can effortlessly read from the linked frames which rules are used to parse the sentence. For example, the utterance in Figure 4 is parsed with the grammar “查询位置 <- 场所 + 哪里(where)” ("query_for_location <- site + where") and “场所 <- 地点 (loc)" ("site <- location").

3.4. Parsing strategy

The parsing process of this parser can be viewed as a merging of semantically meaningful fragments of user utterances. So the parser is a bottom-up one. As the ellipsis phenomena are very common in the spoken language, our parser pays much attention to supporting the resolution of it. It actively maintains a structure of contextual information, waiting to be retrieved at a proper time. A separated module takes on the resolution job and works with the parser seamlessly. The parsing process contains two states as illustrated in Figure 5.
to the ellipsis resolution component, which maintains a variety of domain knowledge representation named “Theme Structure”. According to the theme structure, the ellipsis resolution module attempts to reconstruct the integrated context of that utterance and return it to the parser to continue the interrupted parsing process. Until no more ellipsis is detected and no more merging can be carried on, the parser will finish the parsing and output the candidate results. This process is based on the simple notion that when understanding a sentence the human will not refer to the context unless it cannot be properly interpreted solely.

The parsing process takes advantage of the key ideas of the classical chart parsing algorithm described in [8]. It uses a chart to hold hypotheses that are expecting partial parses to finish the application of the semantic grammars and an agenda to hold the active partial parses that are to be used to expand the hypotheses in the chart. Both chart and agenda are some kind of data structures.

In this parser, a template-filling procedure is adopted to accomplish the parsing. Each rules in the semantic grammar is a template. Every node in the active queue is filled into the templates whose Right-Hand Side (RHS) contains it as an item. Then those fulfilled templates will be evaluated whether the corresponding rules can apply. Because every node is a partial parse of the sentence, that is to say, it covers only a part of the sentence; it is possible that the two partial parses ready to be combined to a bigger parse overlap to some extent. To avoid the overlap, a mask value is used with each node, which indicates what part of the utterance the partial parse covers. If the masks do not conflict and the syntactic constraints are properly met, a new semantic frame will be constructed accordingly. At the same time, a corresponding theme structure will also be generated to record the contextual information. When the parsing without ellipsis resolution ends, the parser enters the ellipsis resolution state as mentioned above. The whole algorithm is given in Figure 6.

```
Initialize the templates and the active queue
While not finished
    While the active queue not empty
        For each node in the active queue
            Fill in the template
            For each template
                Check the mask and the syntactic constraints
                If passed then
                    Construct a new semantic frame
                    Link it to the partial parse
                    Put it into the active queue
                    Construct a theme structure
                    Put it into the dialogue history
    // Enter the state of ellipsis resolution
    Collect the template in need of ellipsis resolution
    For each template in need of ellipsis resolution
        Resolve the ellipsis
        Put it into the active queue
        Output the candidate parse with the highest score
```

Figure 6. Algorithm of the semantic parsing with the ellipsis resolution.

The template of this parser is something like a chart while the active queue acts as an agenda. Thus the template-filling procedure is actually a variation of the traditional chart parsing algorithm. It can be viewed as a Broad-First Search (BFS) in nature.

To pick up the most reasonable interpretation, a parsing score is attached to each semantic label [9]. When several labels are combined to form a new label, the sum of their scores is attached to the new one. Initially, each word of the sentence has a score of 10. Each item obtained from the ellipsis resolution has a score of 5. As mentioned above, it is based on the local preference hypothesis of the human understanding. Thus the parse that has the highest score is considered to be the most preferred one.

4. Conclusions

A semantic parser with support to ellipsis resolution is presented. It uses a variation of the chart parsing algorithm to parse the utterance with a semantic grammar and presents the result with a structure of linked semantic frames. The main aspects of the parser, namely, the semantic grammar, the semantic frame and the parsing strategy are all introduced in detail. To deal with the spoken language, the parser is semantically driven as well as syntactically guided. This is proved to be a good approach with both flexibility and efficiency. The ellipsis resolution is carried on when the bottom-up parsing cannot be properly continued. It employs a domain specific knowledge representation structure to record the dialogue history and reconstruct the context when the ellipsis resolution is needed. With a simple yet effective scoring scheme, the most reasonable parse can be successfully selected from the candidate parses.

Now the parser is lively working in EasyNav, our Chinese spoken language dialogue system for campus navigation task.

5. References