TOWARD A COMMUNICATION DEVICE WHICH GENERATES SENTENCES

Kathleen McCoy, Patrick Demasco, Yu Gong, Christopher Pennington & Charles Rowe
Applied Science and Engineering Laboratories
University of Delaware and AI DuPont Institute

Abstract

Sentence generation will be an integral part of future augmentative communication devices. By employing natural language processing techniques, we hope to enhance the speed, flexibility, and ease of use of current word-based systems. In this paper, we discuss the generation component of our "compansion" system, which expands compressed messages entered by the user into full English sentences. The two phases of this component are (1) the translator, which converts a semantic representation of the message into a syntactic "deep structure" representation and (2) the generator, which takes this "deep structure" and uses a functional unification grammar to produce a complete English sentence. Because of this modular design, the generator is independent of whatever semantic representation is used; thus, it could be easily adapted to other systems (e.g., sign language translation) through the use of a different translator.

Introduction

Sentence generation will be an important component of many communication aids in the future. For instance, it is one component of a system which takes compressed input from the user and generates full well-formed sentences, it could be used in translation devices (e.g., ASL to English), it could be used by a system which improves written English by parsing the writer's prose and then generating better formed sentences when necessary.

In this paper we describe one such generator which is currently used in the "compansion project" at the Applied Science and Engineering Laboratories at the University of Delaware and the AI DuPont Institute. The goal of the project is to allow the disabled individual to input a compressed message containing the content words of his/her intended utterance. The system will take this input, generate a semantic representation using natural language understanding techniques, and eventually generate a well-formed English sentence.

A sentence generator must take a semantic representation of an utterance which is independent of any particular natural language (e.g., English) and translate this representation into an English sentence. We have broken the process into two phases. In the first phase (the translator) the elements in the semantic representation will be replaced by their language specific instantiations. In the second phase (the generator) a grammar of English syntax will be used to order the sentence elements and add proper word endings and syntactic elements so as to output a legal English sentence. In this paper we focus on the generator phase, adding reference to the translator phase in order to point out its novel features.

Background

Input to a generation component is the semantic structure of the sentence to be generated. E.g.,

```
((actor = John)
 (verb = hit)
 (object = ball)
 (instrument = bat))
```

Notice that this information tells what role is played by each content word of the sentence, however, it gives no indication of how the items can be realized in English. In fact there are several valid English realizations of the above information (each of which requires lexical items to be added for syntactic reasons): “John hit the ball with the bat”, “The ball was hit with the bat by John”, “John used the bat to hit the ball”, etc... Of course, one can also construct many non-sentences using the items from the semantic structure (e.g., “John hit ball bat”). The job of a generation component is to enable the generation of the valid orderings while blocking the generation of the invalid ones.

One approach to the sentence generation problem, which is unsuitable for our purposes, is to encode a grammar of English which utilizes the normal syntactic rules and categories of the language (e.g., a sentence is a noun phrase followed by a verb phrase, a noun phrase is a determiner followed by a noun). To use such a component, the translator would have to be responsible for making very language specific decisions about the realization of the semantic input (e.g., the input should be translated as an active sentence whose first noun phrase is John, etc...). Once this is done the input from the translator must be re-ordered to conform to the valid rules of English as embodied by the grammar. This process is much like the reverse of a basic parsing algorithm such as found in [4,1].

The Functional Model

Rather than following the above approach, we have chosen to implement our grammar using the function unification model [2,3]. In this model the functional categories (like those found in the semantic input) may be part of the grammar just as the syntactic categories are. Hence the grammar may specify that a sentence is an actor (which is realized as a noun phrase) followed by a verb, followed by an object, etc... Such a specification allows the translation component to have less language specific information. It still must give actual words and perhaps even identify the category of the roles contained in the semantic structure (e.g., that John can be realized as the noun phrase “JOHN”), however, it need not know any language specific ordering rules (such as the difference between active and passive sentences).

Unification and Generation

In addition to the functional nature of the functional unification model, the model is unique in that it is based on *unification.* Very simply, unification is a formal process which takes incomplete information from two different sources and combines it into one, complete, package of information (provided that no information from one of the input sources contradicts that in the other).

The functional unification model views generation in this way. The two sources of information are (1) the input which contains information about the functional role and lexical realization of the pieces of the sentence to be generated but contains no information about how that information should be ordered or what additional lexical items might need to be inserted in order to make a valid sentence, and (2) the grammar which contains information about several proper orderings of functional features as well as when and where additional lexical information might be needed for purely syntactic purposes.

Notice that both of these sources of information are incomplete alone neither contains enough information to generate a valid sentence. Yet each has what the other needs. By taking them in combination, the valid alternative orderings of the functional features provided by the grammar can be narrowed to the one compatible with the input features, proper lexical (content) words from the input can be inserted into the chosen ordering, and additional (syntactic) lexical items can be added according to the grammar. The unification process combines the two sources of information and the result is a valid English sentence which realizes the input semantic structure.

Conclusion

In this paper we have described a functional unification model for doing sentence generation. The model is attractive since it allows input to the actual generator to be in functional rather than syntactic terms. This allows the translator (which takes the input semantic structure and adds necessary language specific information) to work with minimal knowledge of English syntax. In addition, the model works on the principle of unification which is very attractive for generation purposes. Both the generator's grammar and the input from the translator are viewed as partial descriptions of the sentence to be generated. The grammar contains a specification of legal English sentences but contains no information about any particular sentence to be generated. The input contains specifications for the particular sentence to be generated, but no information about rules of English syntax. The unification process combines this information to generate a valid sentence.

Acknowledgments

This work is supported by Grant #H133E80015 from the National Institute on Disability and Rehabilitation Research. Additional support was provided by the Nemours Foundation.

References

[1] Allen, James. *Natural Language Understanding.* Benjamin/Cummings, CA, 1987.

[2] Kay, Martin. Functional grammar. In *Proceedings of the 5th Annual Meeting*, Berkeley Linguistics Society, 1979.

[3] Kay, Martin. Parsing in functional unification grammar. In B. Grosz, K. Sparck Jones, and B. Webber, editors, *Readings in Natural Language Processing,* pages 125-138, Morgan Kaufmann, 1986.

[4] Winograd, Terry. *Language as a Cognitive Process, Volume 1: Syntax.* Addison-Wesley Publishing Company, Reading, Ma, l983.

Contact

Kathleen McCoy
Dept. of Computer and Information Sciences
University of Delaware
Newark, De. 19716