

An Authoring Tool for Scaling Up in Large-Scale Natural Language Generation

Charles B. Callaway

ITC-irst, Centro per la Ricerca Scientifica e Tecnologica
via Sommarive 18, I-38050 Povo (Trento), Italy
callaway@irst.itc.it <http://tcc.itc.it/people/callaway/>

1 Introduction

We outline an integrated development tool for use in large scale natural language generation. The inverse of parsing, generation starts with information to be communicated (usually in the form of a semantic network) and converts it into text. Generation systems have a wide range of uses, including automatic technical documentation [Reiter et al. 1995], report writing, text customization, scientific explanation generation, intelligent animated agents, tutoring systems, dialogue response generators, and story generation [Meehan 1977].

One of the most significant roadblocks to the widespread adoption of NLG technology is creating the initial linguistic structures necessary to drive NLG algorithms. The larger the text that a system needs to create, the larger the initial data set which must be available. The required knowledge includes lexicon entries for all lexical elements, discourse knowledge, domain knowledge, and rules for discourse planning, sentence planning, revision, lexical choice, pronominalization and surface realization.

The size of the problem was clearly illustrated during our development of STORYBOOK [Callaway & Lester 2001a, 2001b, 2002], an implemented narrative generation system capable of writing 2–3 page stories in the Little Red Riding Hood domain. STORYBOOK was created over an 8 year period, including over a full year to hand-develop the knowledge necessary to generate 3 separate fairy tales.

This software demonstration consists of the presentation of a development environment for large-scale NLG that can reduce preparation time for lex-

icons and discourse plan creation by two orders of magnitude over manual methods. The development environment is written in Java and helps a user construct the diverse types of discourse and linguistic knowledge needed to produce large-scale texts.

During the demonstration, we will describe the various aspects of the user interface and then select a one-page text passage, marking up various lexical elements, rhetorical structures, and other knowledge structures. Additionally, we will invoke the STORYBOOK system on the newly collected information to regenerate the original text passage. Finally, we will demonstrate the STORYBOOK system itself on prepared, parameterizable inputs ranging from fairy tales to newspaper columns, and discuss the time and labor requirements for constructing resources by hand vs. using a development environment.

2 Demo Description

We will show an implemented authoring tool similar to concepts normally intended for knowledge acquisition, but aimed at linguistic structures rather than knowledge bases. This tool is intended for large-scale NLG, *i.e.*, greater than 2 paragraphs. The system conducts an initial analysis to determine which aspects of the document it already knows (*e.g.*, words already in its lexicon) and performs preliminary unassisted markup of the document. The system then enters an interactive mode where the user can correct and add additional annotations. The end result is a complete set of linguistic data structures which can be used by the AUTHOR generation system to recreate the original text.

- **Lexicon Editing:**

The first stage of human-assisted processing is to make additions to the lexicon, where the user creates lexicon entries for a given text for words that were not already contained in the lexicon. A screenshot (Figure 1) shows the user modifying a system-suggested lexicon entry for the verb “to send” at a much quicker speed than creating the lexicon entry by hand.

- **Clause Segmentation:**

The second stage is semi-automatic clause segmentation where the user is again given the opportunity to make corrections. Segmentation is determined by an analysis of both punctuation and discourse markers. Segmentation serves as a basis for the initiation of sentence planning, where semantically structured information has been received from a discourse planner and grouped into sentence-sized chunks. These chunks can later be aggregated by a revision component.

- **Semantics Assignment:**

The third stage allows the user to assign semantic roles to the various aspects of a sentence produced during clause segmentation. Here the authoring tool functions to assist the user in creating sentence planning rules as well as for marking up a particular text with those rules. The sets of possible semantic roles are derived from an associated knowledge base, and thus new lexical items can result in an increase in the coverage of a KB.

- **Immediate Feedback:**

The authoring tool provides assistance to linguistic engineers by both increasing the overall speed at which they can process documents as well as the speed at which they can realize the content of a particular set of knowledge into document form and check it for errors. This type of incremental feedback is valuable for many types of applications where system designers may not be linguists.

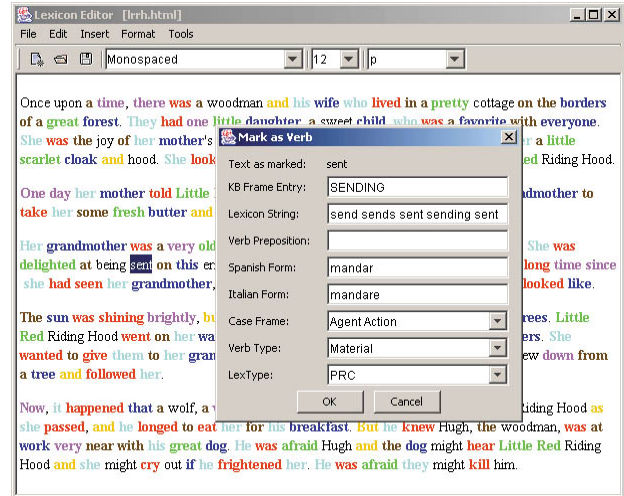


Figure 1: The user modifies the lexicon after automatic preprocessing has been performed.

3 Conclusion

This software demonstration presents a new interactive development environment that vastly improves the time and labor requirements for creating both the linguistic infrastructure and actual text in large-scale NLG. We plan to freely distribute the Java source code for this tool in the near future.

4 References

- Callaway, C., & Lester, J. 2001a. Narrative prose generation. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pp. 1241–1248. Seattle, WA.
- Callaway, C., & Lester, J. 2001b. Evaluating the effects of natural language generation on reader satisfaction. In *Proceedings of the Twenty-Third Annual Conference of the Cognitive Science Society*, pp. 164–169. Edinburgh, UK.
- Callaway, C. & Lester, J. 2002. Narrative prose generation. *Artificial Intelligence*. In press.
- Meehan, J. 1977. Tale-Spin, an interactive program that writes stories. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*. Cambridge, MA.
- Reiter, E., Mellish, C., & Levine, J. 1995. Automatic generation of technical documentation. *Applied Artificial Intelligence*, 9:259–287.