S-Tutor:
A Speech-based Tutoring System

Rolf Schwitter and Md Tawhidul Islam
Centre for Language Technology
Macquarie University, Sydney, Australia
{schwitt | mislam}@ics.mq.edu.au

Abstract. This paper presents S-Tutor, a speech-based virtual tutoring system designed for walk-up-and-use situations in teaching environments. S-Tutor is built on top of the CSLU toolkit, a universal speech toolkit that integrates speech recognition, text-to-speech synthesis, and facial animation. By creating a user interaction similar to that of a quiz game show, S-Tutor attempts to educate students in language technology. The system is not bound to this domain and can be easily configured for new types of quizzes. Teachers can add new topics to a quiz using a text editor that automatically translates expected answers to questions into regular grammars for speech recognition.

1. Overview of S-Tutor

Text-based intelligent tutoring systems are one of the more successful enterprises in artificial intelligence. However, the use of fully speech-based tutoring systems is not well explored and research in this area is likely to advance both the state-of-the-art in tutoring and spoken language dialogue systems. Recent developments in speech recognition and speech synthesis made it possible to incorporate the latest spoken language technologies into tutoring systems.

This favourable trend encouraged us to build S-Tutor, a speech-based virtual tutoring system for interactive quizzes. The S-Tutor system starts an interaction with a spoken welcome message and then provides information about how to use the speech interface. The system then prompts the user to enter the login details into a text field since it is generally difficult for a speech recogniser to recognise unknown user names with high accuracy. After that, the dialogue branches according to the login information either into a subdialogue for students who take a quiz or for teachers who wish to edit a quiz. Each quiz (e.g. *language technology*) contains a number of topics (e.g. *parsing*) with varying degrees of difficulty. The topics consist of a set of questions and answers, hints, and explanations to these questions. Students answer the questions and get marks for each correct answer. If a student does not provide the correct answer, then the system offers a hint or a detailed explanation. Hints are designed to give the student a hand as human tutors do in classroom situations. Equipped with this additional information, the student can reattempt to answer the question but gets lower marks for a correct answer. If the student still does not know the answer, then the system starts playing an explanation. This explanation is written in inverse pyramid style so that the most important content can be played straightaway. The system asks then whether the student would like more detailed information or exit the explanation mode. After all questions of the quiz have been answered by the student, the S-Tutor system plays an appropriate plaudit (if the result is above a specified treshold) and displays a transcript that contains the result of the current session. In the following two sections, we will discuss how a teacher can edit quizzes and show how speech grammar rules for answers are generated automatically.

2. Editing Quizzes

After logging into the S-Tutor system, a teacher can browse the quiz library and modify topics. As already discussed, each topic consists of a set of questions (*wh*-question or yes/no-question), short answers, hints, and explanations. For each new question the teacher has to specify an expected answer, a hint, and an explanation that can be used in the dialogue. To facilitate this specification task, S-Tutor provides a quiz editor (Figure 1):

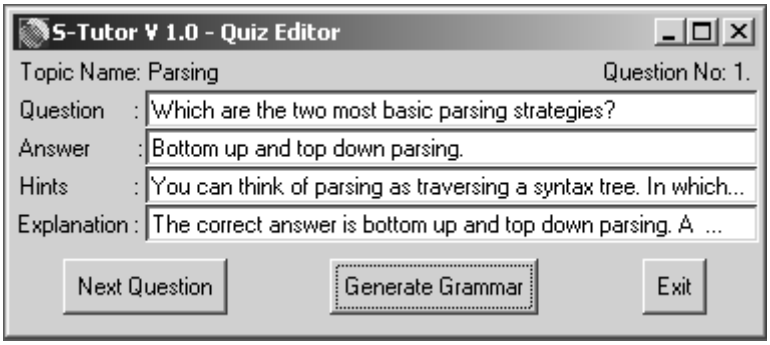

Figure 1: Quiz Editor

This editor allows the teacher to define answers without writing speech grammar rules for the recogniser. Since our domain is similar to the domain of a quiz game show, where the moderator is mainly in charge and the player reactive, the range of answers is limited.

3. Generating Speech Grammar Rules

The CSLU toolkit provides a generic speech recogniser which accepts words and word sequences that are defined by a regular grammar for each dialogue state [1]. The task of the recogniser is to choose the best phonetic path through the finite-state machine defined by the regular grammar. Out-of-vocabulary words are treated by a general purpose "garbage" phoneme model. Grammar rules that describe possible utterances have the following form:

```
$nonterminal1 =  word1 | word2 $nonterminal2 [ word3 word4 word5 ]
```

The `|` character means "or" and *`word1`* or *`word2`* stand for the word that is recognised at that position. The square brackets *`[]`* delimit parts of the grammar rule that are optional. Additionally, special built-in features are available that help to improve speech recognition. For example, two `%%` characters after a word (e.g. `word%%`) indicate that the word will not appear in the recognition results, even if it was recognised. The keyword *`*sil`* is used to recognise silence and *`*any`* is used to recognise anything that does not match the specified vocabulary (including noise).

The S-Tutor system anlayses and translates the expected answer (and the question) specified in the quiz editor into such speech grammar rules and combines them with a template that contains general grammar rules for quiz-specific dialogue situations. Figure 2 shows the grammar rules that have been generated for the example discussed above. Note that only the string associated with the nonterminal symbol `$info` contains the actual answer to the question (in form of a disjunction). In addition, two other grammar rules (`$question` and `$sub`) have been derived from the question. These rules pay attention to the fact that students sometimes repeat the question or paraphrase parts of the question in their

answer. These “content-sensitive” elements of the grammar are then combined with more general “situation-specific” grammar rules (such as `$prefiller` and `$postfiller`) to build the main grammar rule `$answer` for the answer string.

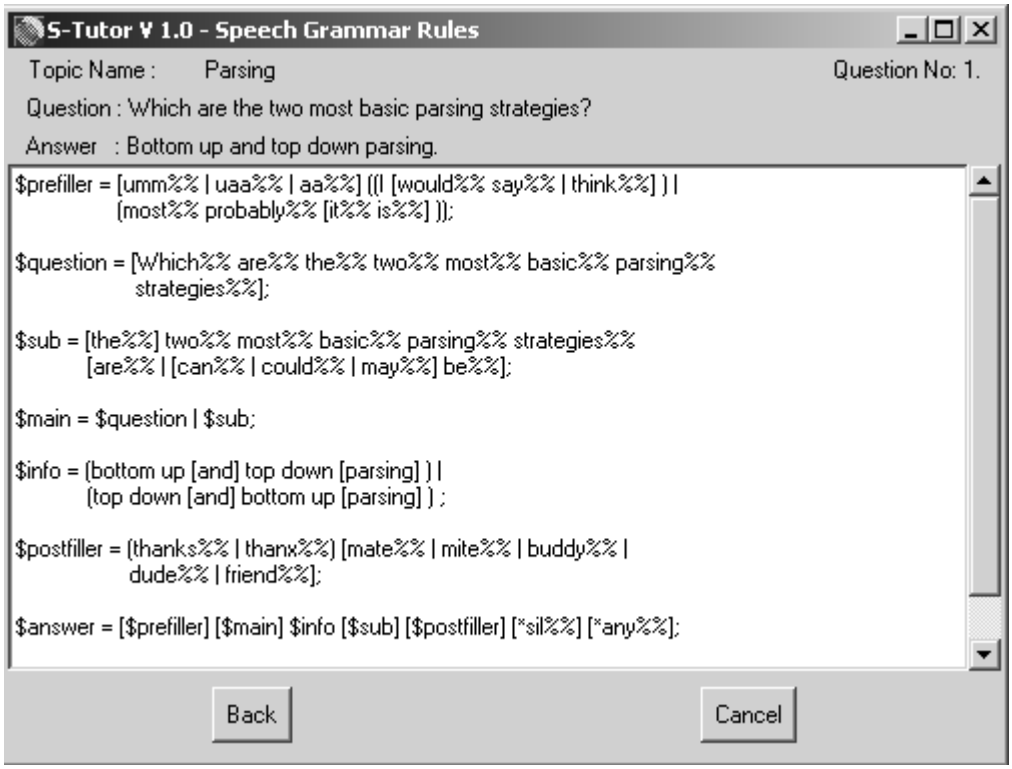

Figure 2: Speech Grammar Rules

4. Conclusion

Although no formal evaluation has been carried out as yet on the use of the S-Tutor system, it is useful to summarize some informal observations. We found that

- tutoring systems can benefit from state-of-the-art spoken language technology,
- especially machine-driven dialogues seem to be very promising,
- speech needs to be combined with visual user interface components,
- good dialogue flow design is the deciding factor for high acceptablity and usability,
- regular grammar rules can be automatically derived from (short) answers.

We plan to investigate how automated checkers, developed in the field of controlled natural languages, may help teachers to edit quizzes effectively and improve processability.

Reference

[1] S. Sutton, R. Cole, J. de Villiers, J. Schalkwyk, P. Vermeulen, M. Macon, Y. Yan, E. Kaiser, B. Rundle, K. Shobaki, P. Hosom, A. Kain, J. Wouters, D. Massaro, M. Cohen. Universal Speech Tools: The CSLU Toolkit, in Proceedings of the International Conference on Spoken Language Processing (ICSLP), pp. 3221-3224, Sydney, Australia, Nov, 1998.