ISCA Archive
http://www.isca-speech.org/archive

6th European Conference on
Speech Communication and Technology
(EUROSPEECH'99)
Budapest, Hungary, September 5-9, 1999

AUTHORING TOOLS FOR SPEECH SYNTHESIS USING THE SABLE MARKUP STANDARD

Johan Wouters *Brian Rundle* *Michael W. Macon*

http://cslu.cse.ogi.edu/tts
Center for Spoken Language Understanding
Oregon Graduate Institute, PO Box 91000, Portland, OR 97291-1000, USA

10.21437/Eurospeech.1999-235

ABSTRACT

In text-to-speech (TTS) synthesis, input text is automatically analyzed. This involves prediction of pronunciation, intonation, and timing at segmental and phrase level. In the design of dialog applications, developers need more control over the text-to-speech conversion. While the automatic analysis is often unsatisfactory, the developer can easily provide hints that improve the synthetic speech. The *Sable markup language*, which has been proposed as a standard for TTS, includes tags to indicate emphasis, speaking rate, phrase breaks, and other properties. We extend this work as follows. First, we describe a graphical editor (GUI) for Sable. An interesting challenge is to find intuitive mappings between the visual representation of the text and the attached markup properties. Next, we discuss the addition of several new markup commands and the implementation in Festival, the TTS platform we use. Finally, we describe our experiences using the authoring tools in a language training project for profoundly deaf children. The authoring tools are made freely available via http://cslu.cse.ogi.edu/tts.

1. INTRODUCTION

In text-to-speech conversion, automatic methods are used to predict pronunciation and prosodic characteristics of the text to be synthesized. Often, semantic and pragmatic knowledge is needed for accurate prediction. For example, the preposition *in* in "Put the kitty in the box," has to be emphasized if the sentence is intended to correct someone's placing the kitty *on* the box.

The Sable markup language has been proposed as a standard for TTS [6, 8]. Sable commands are divided into two categories: *speaker directives* and *text description tags*. Speaker directives include commands to change emphasis, pitch, rate of speech, volume, pronunciation, and phrase breaks. Text description commands indicate structural elements in text (such as a title, sentence, or paragraph) or the function of a block of text (e.g. date, spelled name, URL, etc.).

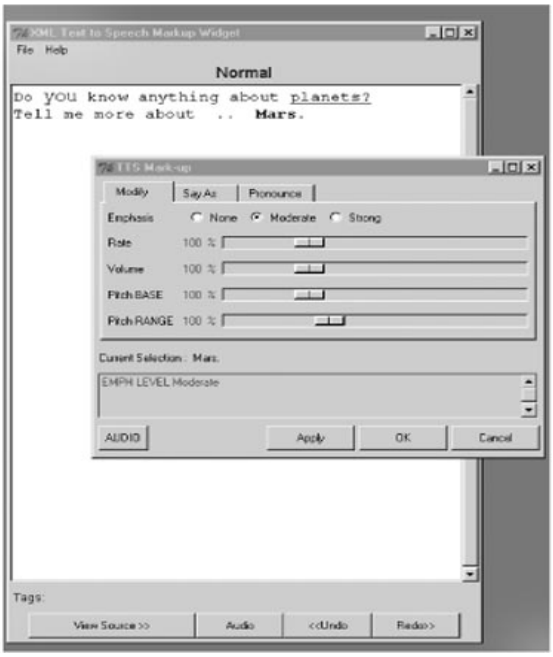
The Sable initiative emphasizes the importance of a TTS standard across platforms and languages. The commands are rather verbose, and it is expected that text generation engines will alleviate the task of specifying tags for a certain document. In this paper we discuss the use of Sable in dialog applications. Typically, the designer of prompts in a spoken dialog creates short phrases or paragraphs, or uses a template to insert contents words into carrier phrases. Our experiences with using TTS in language learning applications (see [5, 4]) show that authors of spoken dialog applications have specific expectations of how the text should be spoken, and are often disappointed with the results of the automatic text analysis. At the same time, these authors can easily provide extra information to help the TTS system interpret the text.

2. A GRAPHICAL USER INTERFACE (GUI) FOR SABLE

We built a graphical user interface to make it easier to annotate text with Sable tags. The tool allows the user to highlight text and attach properties to it. Figure 1 shows an example of the text editor and the customizer that is used to specify the tags. The GUI was implemented in Tcl/Tk.

The graphical interface improves the expressivity of the markup commands because authors can quickly experiment with combinations of tags to reach a desired result. Some problems arise when tags are potentially in conflict. For example, if a pronunciation tag spans more than one word, tags on a subset of these words should be disallowed, since they should apply to the substituted pronunciation and not to the original words. This conflict is cur-

*This work was supported in part by NSF CHALLENGE grant CDA-9726363, "Creating Conversational Agents for Language Training," and by Intel Corporation.



Do YOU know anything about planets?
Tell me more about .. **Mars.**

Emphasis ○ None ● Moderate ○ Strong
Rate 100 %
Volume 100 %
Pitch BASE 100 %
Pitch RANGE 100 %
Current Selection : Mars.
EMPH LEVEL Moderate

Figure 1. Markup editor with customizer for annotations.

rently not addressed in the Sable standard.

Representing the TTS annotations visually is an interesting challenge. In word or HTML editing, properties are attached to text to create a specific graphical effect. With TTS markup the intended effect is auditory; while it is up to the interface to express it graphically. We found little research on this topic, although many examples of expressive typograpy can be found in novels, magazines, and web documents.

A paper by Henton *et al.* explores color, width and height of letters to express five basic emotions [1]. However, when more TTS anotations need to be expressed graphically, much more careful choices need to be made. In addition to finding intuitive mappings between the typography of the text and how it should be spoken, care has to be taken to keep the document readable and attractive.

We considered the following properties of fonts: height, width, color (foreground and background), style (capitals, italics, boldface and underline), and font family. In addition, extra characters or icons can be inserted in the text. Table 1 summarizes our mappings between markup commands and graphical properties. Obvious choices are: font size for *loudness*, boldface for *emphasis*, inserted text or icons for *breaks* and *audio inserts*.

Loudness	Font Size
Emphasis	Boldface
Speech Rate	Font Width
Pitch	Underline
Sayas	Red
Pronunciation	Red
Audio Insert	Icon Insert
Break Insert	Icon or ... Insert
Speaker or Language Change	Icon Insert

Table 1. Mappings from markup tags to graphical properties.

More problematic mappings, in our opinion, are *pitch* and *pronunciation* or *speaking mode*. Markup commands can change the average pitch, the pitch range, or can impose a pitch contour (see Section 3.). We have chosen to represent this by underlining text where the pitch is to be modified. Pronunciation and speaking mode changes are reflected by coloring the fonts in red. For all the mappings, we have chosen to let the typography reflect which type of markup tag has been attached, while the exact parametrization of the tag (e.g. level of emphasis, percentage increase of speech rate) is hidden. The user can find out more of the detail by "mousing over" the annotated text, or by opening the customizer, which will reflect the exact tags.

3. NEW MARKUP COMMANDS

The Sable standard consists of a fairly large set of markup commands that was based in part on several existing tag sets [6]. Whereas these commands are certainly needed, we discovered that they are often not powerful enough for language training applications. The XML (Extensible Markup Language) formalism makes it easy to add new commands to Sable and such extensions are in fact invited by the Sable consortium. We have added several commands, some of which we hope will also be useful for dialog design in general. The additions presently do not have the "X-" prefix as requested by the authors of Sable.

3.1. Fine Level Control

The standard `PITCH` and `VOLUME` commands in Sable allow to specify average values, but don't give the user control over a precise intonation or loudness contour to be followed. Hence we added a `CONTOUR` attribute as in:

```
<PITCH CONTOUR='0.0 80; 0.4 120; 0.6 100; 1.0 80'>.
```

The coordinates in the contour are time (relative to the tagged text) and pitch (in Hz). A similar contour attribute was defined for the `VOLUME` tag. Support for specifying contours can be provided easily using the graphical interface of Section 2.

An important requirement in language training is the ability to specify text phonetically and to change the durations of the segments independently. The Sable standard advocates the use of Unicode IPA (International Phonetic Alphabet) characters because it is an international standard. However even if the characters are displayed correctly on a computer screen, it is cumbersome to type them or represent them in a computer program. Hence we have adopted the use of an ASCII phonetic alphabet, called Worldbet [2]. We believe this alphabet could be a good candidate to be adopted in standard Sable.

In addition, we have created support for the specification of segmental durations as in:

```
<PRON SEGDUR='s 50; ei 100; b 40; l= 70'>
```

where the durations are in milliseconds. The durations and phonemes can be conveniently specified in the drag-and-drop paradigm of the graphical interface.

3.2. Speaking Modes

Authors also like to use higher level commands. The SAYAS command in Sable allows a user to indicate the function of a text fragment, such as a date, the spelling of a name, a computer variable, a phone number, a URL, etc. It is left up to the TTS engine to translate the text fragment into a string of words. However, those words often also need to be spoken according to a specific rhythmic and intonational pattern. High level commands thus need to be integrated correctly with the existing prosodic prediction algorithms of the TTS engine. Even "simple" speaker directives such as *duration stretch* or *emphasis* become complicated if more natural speech rate modification and emphasis are desired.

Our TTS platform of choice is Festival, a highly modular and extensible synthesis system freely available from the University of Edinburgh [7]. By building on the existing modules in Festival, we have implemented new SAYAS commands, such as *slow* and *fluent* spelling, and *syllabified* articulation of words (as in "syl – la – bi – fy"). Acceptable prosody is achieved via heuristic rules which involve assigning Tobi labels, inserting silence segments, and normalizing syllable durations.

Through use of the SRC command, paralinguistic sounds such as coughs, clicks, yawns etc. can also be inserted from audio files. Other speaking modes we would like to create are hyper-articulation, child-directed speech or *motherese*, and emotional cues. Supporting these new speaking modes will place higher demands on the waveform generation part of the synthesizer. Voice characteristics such as breathiness, increased pitch, variation in speaking rate, and hyper-articulation of phonemes are not well modeled by today's speech synthesizers. We believe increased use of TTS markup can be a catalyst for research in this direction.

4. EXPERIENCES FROM USING SABLE AT THE TUCKER-MAXON ORAL SCHOOL

The work described in this paper fits in a larger project for teaching spoken language to profoundly deaf children. Our partners in this project are the Tucker-Maxon Oral School in Portland, Oregon, and the University of California – Santa Cruz (see [3, 4, 5] for more details about this project). The children at the school have cochlear implants or use amplification devices to compensate for their hearing loss. In class, emphasis is placed on improving their understanding and production of speech. The use of spoken language technology is explored to enhance the children's learning. Teachers make computer-aided language lessons involving speech recognition, speech synthesis, facial animation and images. The applications are self-guided so that the children can practice what they have learned from the teacher, while a *conversational agent* acts as a personal tutor.

In this section we give some practical examples that highlight the importance of TTS markup in a real-world application. Formal assessment procedures are an integral part of the project, but results are not available at this point. Instead, we describe several areas of spoken language teaching and the contributions made by TTS markup in those areas.

4.1. Comprehension

Several of the applications developed at Tucker-Maxon focus on understanding spoken language. The exercises include images and spoken directives such as "click on the bananas," "put the kitty in the box," or "tell me more about Mars." When the child's response is incorrect, the conversational agent repeats the directive with specific emphasis on the key word: "click on the **bananas**," "put the kitty **in** the box," "tell me more about **Mars**."

The fact that a sentence is intended as a repetition or a correction cannot be predicted from the sentence itself. Hence most automatic TTS methods will realize the above sentences in exactly the same way when they are repeated, which can be very frustrating for both the teacher that develops the application and the child that engages in it. With the right authoring tools, however, the experience is much more positive.

4.2. Speech production

An important aspect of spoken language training is the correct perception and articulation of all phonemes and syllables in a word. Several applications were developed that focus on minimal pairs like "**b**ee" and "**p**ea", or "**s**ee" and "**sh**e". Other applications focus on the formation of verbs, e.g. "she's sleep**ing**" if the child said "she sleep", or "under**stood**" if the child said "understanded".

For these applications, it is necessary that the synthetic voice can emphasize or hyper-articulate the important phonemes. This is currently supported via markup commands specifying segmental durations. Eventually, we want to develop a *hyper-articulation* speaking mode.

In exercises on multi-syllabic words, the speaking mode *syllabify* is used. In this mode, the syllables are spoken slowly and rhythmically, as in "ar – ti – cu – late". If the child skipped a particular syllable, this syllable can be emphasized during syllabification.

4.3. Conversation and story telling

In conversational exchanges, teachers accentuate rising intonation for certain questions, falling intonation to indicate the end of a speaker's turn, etc. To practice conversational skills with the animated agent, the synthetic voice should have the same ability to exaggerate a specific rising or falling intonation contour. The markup commands which we have described support this.

In another application developed at Tucker-Maxon, children listen to a short story told by the agent, accompanied with pictures on the computer screen. While most TTS engines predict intonation and timing per sentence, in a longer paragraph it is important to maintain a certain "flow", and to place appropriate stress on new elements, while deemphasizing what is already known. The markup allows developers to do this difficult processing beforehand.

Instead of synthetic speech, human recordings can also be inserted in the applications. Still, synthetic speech is often preferred because the voice of the conversational agent should remain consistent, alignment of the animated face with recorded speech is difficult, and because applications can create new sentences at run-time. However, it should be possible to mark up text by showing a human example, instead of describing a number of commands. For this purpose, we plan to integrate pitch-tracking and phonetic alignment tools in the graphical interface.

5. CONCLUSION

The tools we have started to develop have been received very enthusiastically by the teachers involved in the project. Eventually, the tools should also be used by the children to experiment with the speech of the conversational agent. The teachers' requirement for speech synthesis is that "the synthetic voice should be able to do what teachers do." This includes controlling intonation, articulation, and timing in order to emphasize (exaggerate) any speech event. Challenges with the current tools are to visually represent marked up text, and to improve the synthetic speech to include wider ranges of speaking modes such as emotion, hyper-articulation, and story telling.

REFERENCES

[1] C. Henton and P. Litwinowicz. Saying and seeing it with feeling: techniques for synthesizing visible, emotional speech. In *Proc. of the Second ESCA/IEEE Workshop on Speech Synthesis*, pages 73–76, September 1994.

[2] J. Hieronymus. ASCII phonetic symbols for the world's languages: Worldbet. ftp://speech.cse.ogi.edu/pub/docs/worldbet.ps, 1994.

[3] NSF Challenge Grant: Creating Conversational Agents for Language Training. http://cslu.cse.ogi.edu/tm.

[4] R. Cole et al. Intelligent animated agents for interactive language training. In *In STiLL: ESCA Workshop on Speech Technology in Language Learning*, Stockholm, Sweden, May 1998.

[5] R. Cole et al. New tools for interactive speech and language training: Using animated conversational agents in the classrooms of profoundly deaf children. In *ESCA-MATISSE ESCA/SOCRATES Workshop on Method and Tool Innovations for Speech Science Education*, London, UK, April 1999.

[6] R. Sproat et al. Sable: a standard for TTS markup. In *Proc. of the International Conf. on Spoken Language Processing*, Sydney, Australia, December 1998.

[7] The Festival Speech Synthesis System. http://www.cstr.ed.ac.uk/projects/festival.

[8] The Sable Consortium. http://www.cstr.ed.ac.uk/projects/sable.html.