Information Transfer from Dialogue Response Generation to Speech Synthesis

Paul C. Bagshaw

Orange Labs - France Telecom
Cesson-Sévigné, France.

Application Context

A dialogue system generates responses to information queries. In the simplest case, the response takes a textual form, just a sequence of words. The response may be then fed into a speech synthesis system to convert it to an audio signal. In this case, the synthesis system transforms raw text to speech. In doing so, the synthesis system may perform syntactic and semantic analyses of the incoming text in order to determine the performance structure (pause placement, prosodic word grouping, inter-word break indices) of the outgoing utterances. Note, however, that when generating a response, the dialogue system adheres to strict grammatical constructs, it holds semantic information on the words it employs, and it has knowledge of the flow of the discourse. It thus seems inefficient to pass only a sequence of words from a dialogue response generation system to a speech synthesis system when syntactic, semantic and discourse information already exists on one side of the interface and is required (or may be used) on the other side.

Limitation of using SSML for Information Transfer

SSML 1.1 provides a mechanism to enrich the transfer of information between the dialogue and speech synthesis systems, however:

- The <p> and <s> elements add little to what may already be deduced from correct use of punctuation (which is easily imposed in this context) and the scope of these elements is too broad to describe sentential syntactic structure.
- The <emphasis> element is more informative in that it may be an indication of how to realise the dialogue's focus, but it is somewhat unsuitable for distinguishing discourse theme (topic) from rheme (comment, focus) per se. A dialogue response generation system does not usually hold information about degrees of emphasis on particular words. A speech synthesis system may use theme/rheme in combination with syntactic structure to derive such information.
- The *role* attribute of the <token>/<w> element may be used to provide word-level grammatical tags to the synthesis system. However, the set of grammatical tags used by the dialogue response generation may be incompatible with that used by the synthesis system.

Constraints on the Interface between Dialogue and Synthesis Systems

Existing (non-research) dialogue systems rarely hold explicit information about the exact syntactic structure or the grammatical category of *all* words in their generated responses to queries (they tend to use template sentences and fill in the holes). Furthermore, all synthesis systems already have a mechanism to determine syntactic structure and grammatical categories for their own needs from just a sequence of words (text only). Thus it is rarely possible or unnecessary to transfer *complete* syntactic/grammatical structure from dialogue to synthesis systems.

Attention should be placed on the **requirement** to transfer information that a) exists in current dialogue systems, b) may be exploited by synthesis systems to improve/control the speech rendered, and c) may not be readily derived from text alone by a synthesis system. There are essentially two areas that fit these criteria today. The first is *discourse* information (e.g. speech acts, theme/rheme localisation) and the second is information leading to *word disambiguation* (e.g. localisation of named entities, grammatical tags – to distinguish forms with differing pronunciations, "record" – semantic roles – e.g. "base" a fish or a musical instrument – and preposition/verbal phrase attachments).

Developers of speech synthesis systems have already developed (at great cost) complex and integrate techniques to determine the phonetic sequences and performance structure of utterances from text. These techniques rely heavily on a *fixed* set of grammatical tags that are used coherently between their internal syntactic grammars, morphological analyses, pronunciation disambiguation procedures, prosodic models, etc. When communicating with the components of these synthesis systems to control their behaviour, it is necessary to use the same *fixed* set of grammatical tags. Any interface with a speech synthesis system **MUST NOT** require the set of grammatical tags used by it to change. Not to satisfy this constraint would unacceptably impose costly redevelopment of any TTS system. Similarly, dialogue systems providing information to speech synthesis systems will have their own internal representations for the information. The interface must not interfere with them either.

Proposed Extension of SSML with Structural Information

A new <group> element, with a required attribute *type* and an optional *role* attribute can largely enhance the mechanism for transferring information between dialogue and speech synthesis systems. Possible values for the *type* attribute can be "syntax", "prosody", and "discourse" (these values would need to be standardised). The *role* attribute can be a list of QName values (these would *not* need to be standardised), as it is on the <token>/<w> element, and can be used to convey, for example, syntactic forms, levels of prosodic phrasing, and speech acts.

In the case of an interface between dialogue and synthesis systems, attention should be placed on such an element being able to transfer discourse structure (particularly theme/rheme localisation, which can help place emphasis on prosodic words), and to delimit named entities (which can help word disambiguation).

The proposed *type* "prosody" is secondary to the dialogue/synthesis use case, but it illustrates the intended flexibility of the <group> element, and it provides a mechanism for the occasional need to control directly the prosodic structure of utterances (as opposed to indirect control via the specification of syntactic and discourse structure).

Interface for Differing Sets of Grammatical Tags

The interpretation of the values of a *role* attribute is problematic when used across different systems, wherever it is used (<token>/<w> or <group>). Take the case when the *role* attribute is used to convey the grammatical tag of a word. The format of the grammatical tags may differ between systems. Furthermore, even when the format is identical, the interpretation of the grammatical tags attributed to a word may also differ between systems.

The definition of a set of grammatical tags is fundamentally tied to its use. A dialogue system may attribute a word with a list of grammatical tags based on the words *function*. A noun (e.g. "drink <u>water</u>") can function as an adjective (e.g. "<u>water</u> sports"), and adjectives (even those

that come from nouns) can function as verbs (e.g. “water the plants”). A grammar declared for a dialogue system will be written in a way that exploits the *functional* nature of the tags (thus allowing, for this example, the word water to occur in these three constructs by attributing a tag of noun/adjective/verb). In contrast, a synthesis system may use a set of grammatical tags based solely on a words *form* (e.g. water is tagged as a noun only), since this is largely sufficient in determining word pronunciation (phonetic transcription, disambiguation of heterophonic homographs, lexical stress patterns, etc.).

It is important to note here that even though system A can say X is a noun or an adjective, system B does *not* want to interpret X as being a noun or an adjective when that information comes from system A (to do so would lead to erroneous behaviour of system B). That is because system A describes words by their grammatical *function* and system B wants a description of words in terms of their grammatical *form*. [[Analogy: you can call a spade a spade, but a spade is not a spade when it comes from a pack of cards.]]

The interface between dialogue and speech synthesis systems is **required** to provide a link between differing *interpretations/meanings* of the values of the *role* attribute. The interface also **requires** a link between the interpretations/meanings and the *values* of the *role* attribute on each side of the interface.

Proposed Three Components for “role” Mapping

The mapping of a set of grammatical tags issued from a dialogue system to that required by a synthesis system may be defined from three components: (1) a description of the source (dialogue) tag set; (2) a description of the target (synthesis) tag set; and (3) a description of the transliteration between the two sets. The source (1) and target (2) resources define the fixed sets of grammatical tags used by the two systems sitting on either side of the interface in terms of features. The transliteration (3) then defines the link between them in terms of these declared features.

The source and target tag sets (1) & (2) can be defined by a common XML format.

For example,

```
<set xmlns:dialog="http://www.example.com/dialog">
<class name="tag" wildcard="*">
    <cat name="noun" value="N">
    <class name="number" wildcard="*">
          <cat name="singular" value="s"/>
          <cat name="plural" value="p"/>
    </class>
    </cat>
    …
</class>
…
</set>
```

The above defines *role*="dialog:Ns" as a singular noun and *role*="dialog:Np" as a plural noun.

Similarly,

```
<set xmlns:tts="http://www.example.com/tts">
<class name="Gram" wildcard=".">
    <cat name="Nouns" value="ZZ">
    </cat>
    …
</class>
…
</set>
```

defines *role*="tts:ZZ" as nouns (with no sub category available to distinguish singular from plural)

The transliteration (3) may then be defined in terms of class and cat names.

```
<equiv>
    <src namespace="dialog">
          <class name="tag" cat="noun">
            <class name="number" cat="*"/>
          </class>
          …
    </src>
    <dst namespace="tts">
          <class name="Gram" cat="Nouns"/>
          …
    </dst>
</equiv>
```

This defines equivalence sets. The wildcard value enables a compact representation for many-to-one and one-to-many mappings. The outcome of these is a link from dialog:Ns and dialog:Np to tts:ZZ. The internal representations for the dialogue and synthesis systems need no modifications what-so-ever and the mapping between the two is plainly made.

Pragmatic Issue

Any implementation of an enhanced interface between a dialogue response generation system and a speech synthesis system must be efficient (in terms of computational complexity) relative to undertaking linguistic analyses on the raw text. The addition information supplied is expected to be pertinent to rendering the speech; i.e. it should be an improvement to that which can be determined from text alone.