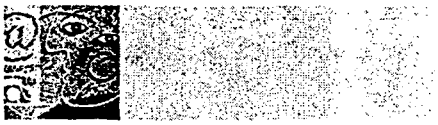


NORTEL
NORTHERN TELECOM

# Recognition of spontaneous speech

**Peter Stubley**

Nortel OpenSpeech

**Abstract**

Current speech recognition systems are capable of performing complex tasks for co-operative users by determining their requirements through a conversation. Most systems have been constructed without attempting to accurately model spontaneous speech. Some components, such as the parser, can be easily made robust to some of the artifacts of conversational speech. Others, such as the pronunciation models, simply ignore the possibility that incomplete words can occur. This results in some recognition errors, and may cause the application to begin to perform the wrong the action. Typically, however, the next several conversation turns can identify and correct the error.

This talk gives a brief overview of state-of-the-art of spoken language systems and describes how some of the components are affected by artifacts of spontaneous speech.

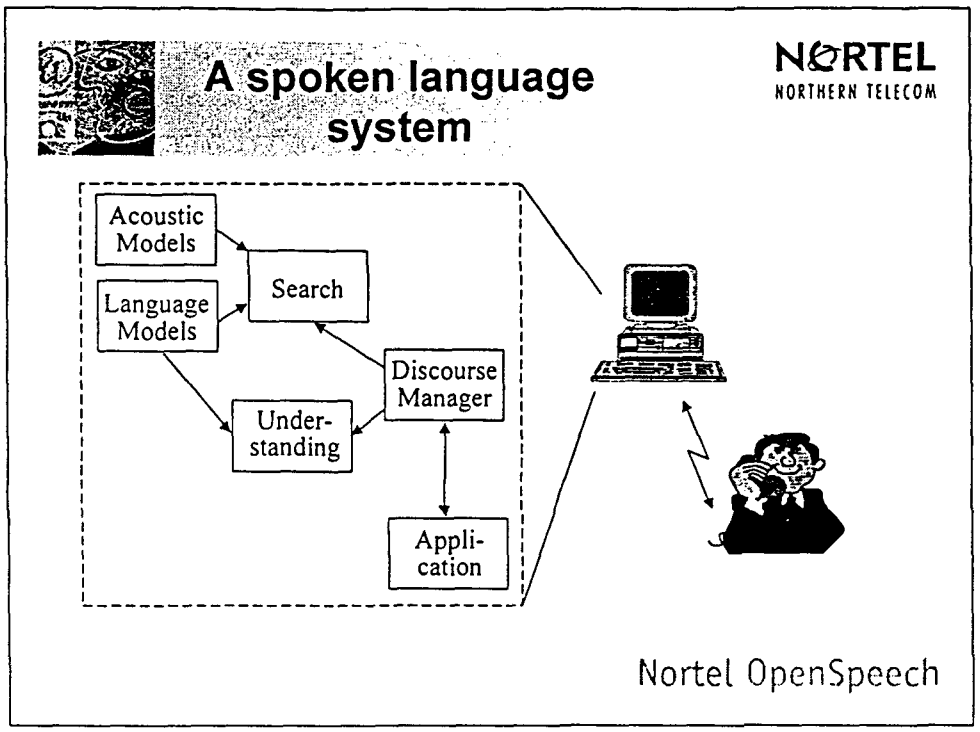
Large bodies of accurately transcribed spontaneous speech are required to learn the properties of spontaneous events.

# Outline

NORTEL
NORTHERN TELECOM

- **Components of a speech recognition system.**
- **Modeling speech:**
  - acoustic models.
  - pronunciation models.
  - language models.
- **Natural language understanding.**
- **Discourse management.**
- **Effects of spontaneous speech.**

Nortel OpenSpeech




The search engine matches the user's speech to the most likely path through the search graph. The search graph is specified by the acoustic models, and the language model (where the language model includes pronunciation). The most likely path corresponds to a word sequence. The matching is usually accomplished using a coarse quantization of the speech spectrum and some variation of the Viterbi algorithm (dynamic programming).

This word sequence is passed to the discourse manager who in turn passes it to the understanding component. The meaning is extracted and returned to the discourse manager. The discourse manager takes the appropriate action and tells the user the result.

## Modeling speech

NORTEL
NORTHERN TELECOM

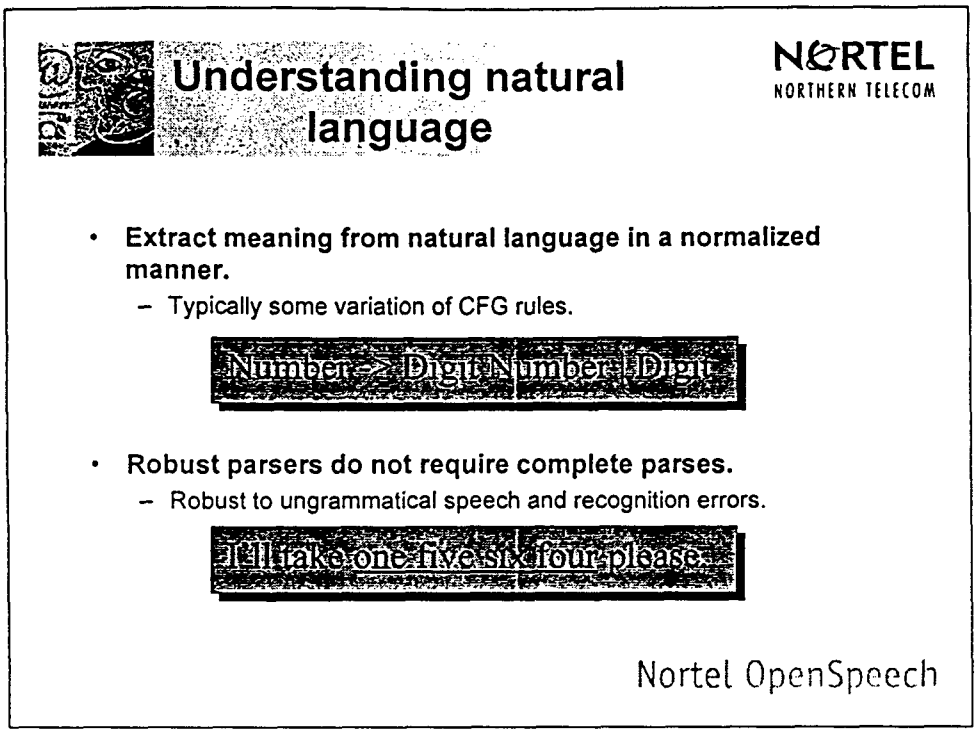
- **Acoustic models.**
  - Model speech sounds, typically phone-based models.
- **Pronunciation.**
  - How words are pronounced, typically concatenations of acoustic models as defined by the lexicon.
- **Language model.**
  - How words may be connected together, typically statistical for large applications.

Nortel OpenSpeech

The acoustic models are usually some variation of hidden Markov models (HMMs). The state sequence helps to model the quasi-stationarity of speech with discrete jumps from one type of statistics to another (such as a transition from a fricative to a vowel). Each acoustic model typically corresponds to a phone in a particular context. Acoustic models are trained from a large corpus of transcribed data.

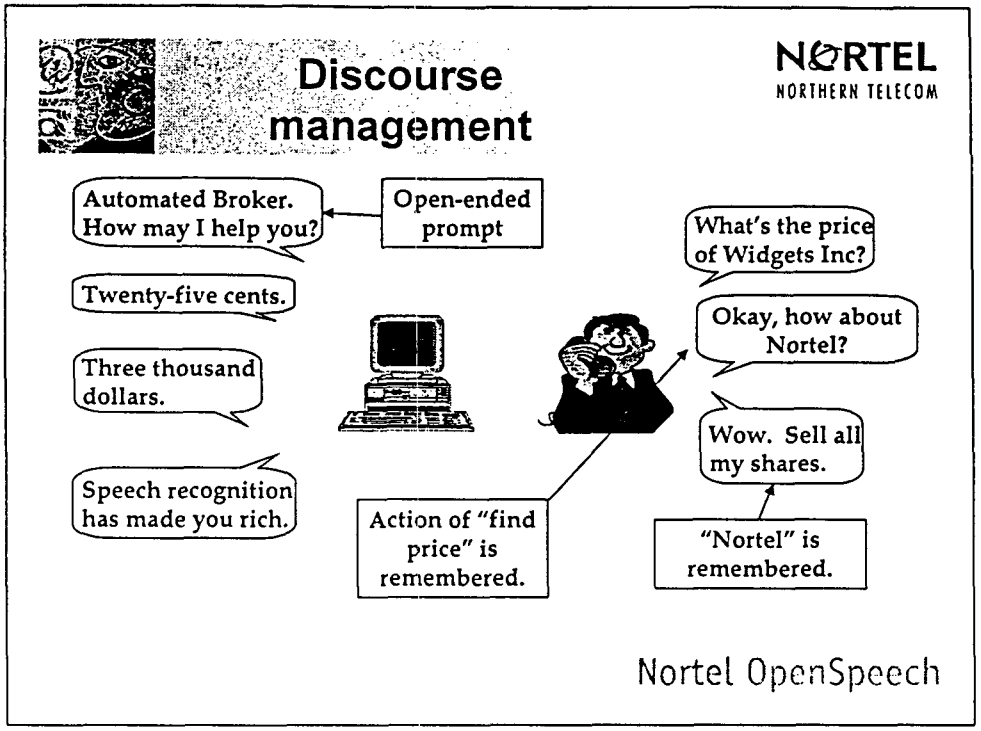
Words models are constructed by determining the pronunciation of each word in a lexicon. The string of phonemes is mapped to a sequence of acoustic models - the resulting chain of models becomes the model for the word. With this approach, models can be constructed for each word without actually having training data specifically for that word.

The language model describes how words may be connected together. The most common language models for large applications are purely statitistical, with the most common ones being the defined by the previous several words. Bigram models give the probability of each word depending on the previous word and trigram models give the probability of each word depending on the previous two words. Language models are trained on large corpora of text as well as the transcribed data used to train acoustic models.




The NLU component attempts to extract the meaning from the recognized word string. This is typically accomplished by matching CFG rules to build parse trees. For example, a number can be represented by a digit followed by another number or simply a digit by itself. Recursively applying this rule can represent any number.

In most cases, a robust parser is used. A robust parser is similar to a normal parser, but does not require a complete parse to succeed. Instead, a forest of parse trees is found that represent the highest level rules that could be found. With this approach, words that do not fit any rule (because of ungrammatical speech or misrecognitions) are simply left out of the parse, and the parser returns what it can find. For the above example, since the parser only has a rule for a number, it will extract only "one five six four" and ignore "I'll," "take," and "please."




The discourse manager is responsible for carrying on the conversation with the user. Ambiguous or unclear information is clarified by the DM throughout the dialogue by:

- asking the user what they meant or prompting for the missing information.
- confirming the most likely interpretation.

In a spontaneous dialogue, the discourse manager is required to infer things based on the history of the conversation. For example, if the previous request was for the price of a stock and the subsequent request gives only a stock, the most likely interpretation is that the user also wants the price of the subsequent stock.

The discourse manager also interacts with the actual application, such as a data base, voice mail/e-mail server, etc. Thus, for each application, the discourse manager must understand the requirements of the application, how to express the user's request to the application, and how to interpret the response from the application.

## Effects of spontaneous speech

NORTEL
NORTHERN TELECOM

- **Phrases may frequently be ungrammatical.**
  - Sentence fragments, sloppy usage.
- **Examples:**
  - Reserve tickets two people ten o'clock show.
  - Gotta go now.
- **Natural language may also include idiomatic expressions.**
- **Examples:**
  - "That's cool!" likely has nothing to do with temperature.

Nortel OpenSpeech

People frequently speak in sentence fragments or use innovative constructions for effect. The meaning is clear, but no self-respecting grammar textbook would permit it. Robust parsing solves many of these cases.

Although the number of rules in the grammar can be increased to include all of the possible variations, a number of new problems will be introduced:

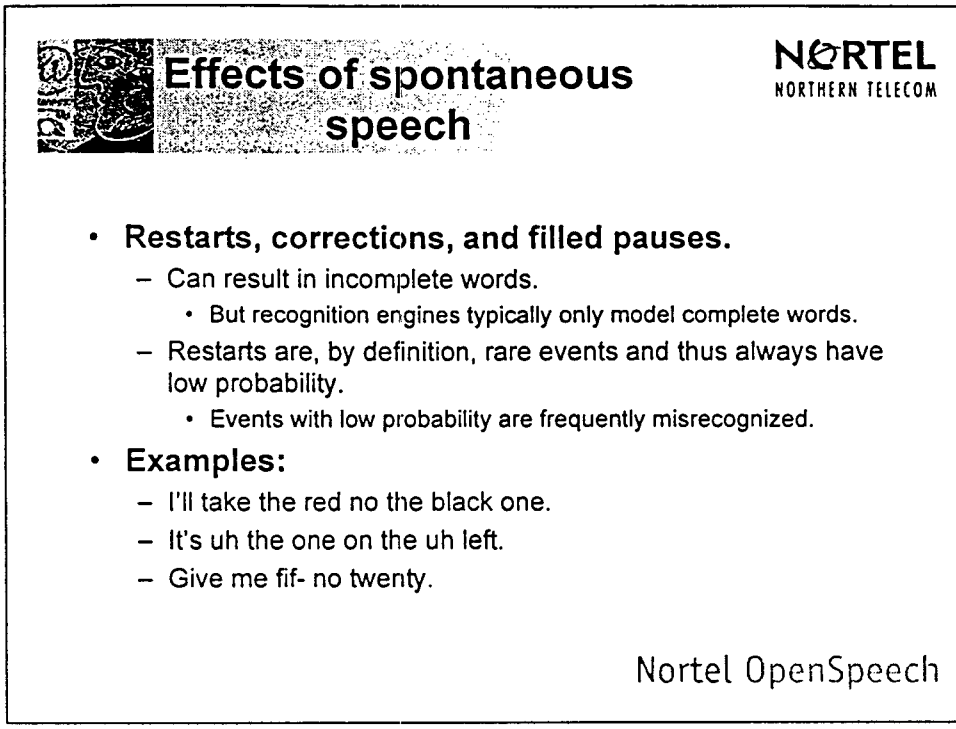
• computational complexity of a large number of rules.

• maintaining and debugging the rules is very difficult. New rules will likely have unforeseen consequences and may conflict with existing rules.

• determining and writing all of the rules in the first place is time-consuming.

Writing rules is similar to writing a program in any other computer language. Unless they are carefully designed, large programs are brittle and difficult to make bullet-proof.

Natural language also includes many other effects, such as idiomatic expressions and puns. Some of these will change with time. Extracting the meaning from expressions containing these can be difficult. In practice, particularly given the state of today's voice synthesizers, people realize that they are speaking to a machine and will adjust their language accordingly. The sentence fragments will remain, but some of these effects will be much rarer than they will be in conversation with another human. Thanks to Star Trek, people are used to computers interpreting idiomatic expressions literally.




Filled pauses are perhaps the easiest to deal with. Words representing the filled pauses (such as ah, um, uh) can be added to the recognizer's lexicon and incorporated into the language model.

Restarts and corrections that do not include incomplete words also affect only the language model. The presence of an indicator word or phrase followed by a phrase similar to the correction can be incorporated in the language model. The biggest difficulty is that statistical language models typically have limited histories, and thus the fact that the following phrase is similar to the preceding phrase is usually lost.

Incomplete words are the most difficult to handle. The lexicon only contains complete words. Incomplete words can be permitted but this typically results in an explosion of the number of paths in the search graph and many incomplete words will be easily confused with other words. Including them everywhere is likely to make things worse, both in terms of accuracy and speed.

## Effects of spontaneous speech

NORTEL
NORTHERN TELECOM

- **Spontaneous events are not completely random.**
  - For example, breaks for incomplete words are likely to occur only at (or near) syllable boundaries.
- **They are almost always rare events in any given word sequence.**
  - Unlikely events are difficult to model accurately.

⇒ *Accurately transcribed data are required to understand these events.*

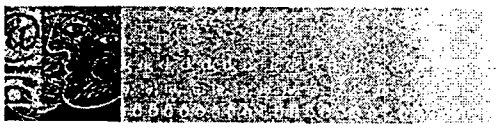
Nortel OpenSpeech

Spontaneous events are not completely random. They are also affected by the speaker's style - some people have few filled pauses, others many.

Spontaneous events often appear to be mostly uncorrelated with the actual word sequence. As a result, they do no fit well into typical statistical language models. Since they are not modeled well by the statistical model, they always appear to be unlikely.

The probability of spontaneous events at any particular place is low; the probability of spontaneous events occurring during any conversation is high.

With large bodies of accurately transcribed data, models that attempt to incorporate spontaneous events can be constructed. The more accurately spontaneous events can be modeled, the better they will be recognized.



NORTEL
NORTHERN TELECOM

# Uh, mis- no recognition of uh spon- spontaneous speech

Nortel OpenSpeech

## Bibliography and references


• Deller, J.R., Proakis, J.G., and Hansen, J.H.L., Discrete-time processing of speech signals, MacMillan, 1993.

• Lee, C-H, Soong, F.K., and Paliwal, K.K., Automatic speech and speaker recognition, advanced topics, Kluwer Academic Publishers, 1996.

•O'Shaughnessy, D., Speech Communication, Addison-Wesley, 1987.

• Rabiner, L.R., "A tutorial on hidden Markov models and selected applications in speech recognition," Proceedings of the IEEE, Vol. 77, No. 2, February 1989, pp. 257-285.