



Inference Communications

White Paper

Natural Language Speech Recognition Solutions
Can I build them using SRGS?

For more information contact
info@inferencecommunications.com

3/09/2008





*Inference is a world leader in **natural language** speech recognition technology and has developed a range of configurable, pre-packaged, natural language speech solutions.*

1 Introduction

'Natural Language' in the context of a speech recognition solution refers to the ability of the caller to the system to be able speak naturally and be understood. There are a couple of ways in which a natural language solution can be built; one method is to use a Statistical Language Model (SLM) approach to building the grammar files. Another is to use context free grammars based on the Speech Recognition Grammar Specification (SRGS) to build mixed initiative natural language applications.

So which is better? Well read on, and you'll see that either approach will work, but the time and money associated with building the grammar will vary enormously.

2 Natural Language

Speech recognition applications can often be divided into two groups based on the way the caller interacts with the system :

1. Directed Dialog (single slot) systems

These are speech systems that guide a caller through the application and only requiring them to enter single words or short phrases in response to prompts.

Example: *In an airline ticketing speech application, a user would be prompted with "Which city would you like to fly to?" and the caller would respond with a simple response e.g. "I want to fly to Paris".*

2. Mixed Initiative (multi slot) systems

These are speech systems that allow the caller to interact using natural language input.

Example: *In an airline ticketing speech application, a user could say "I want to fly to Singapore tomorrow in business class". And be understood. These mixed initiative applications are commonly referred to as 'Natural Language applications'.*

3 Grammars

A speech recognition grammar is a set of word patterns, and tells a speech recognition system what to expect a human might say. For instance, if you call a directory assistant application, it will prompt you for the name of the person you would like to talk with. It will then start up a speech recognizer, and load a speech recognition grammar. This grammar contains the names of the people in the directory, and the various sentence patterns callers typically respond with (e.g. "I would like to speak to John Smith please").

The speech recognition industry has defined a number of 'standard' formats for the grammar files used in speech recognition systems.

1. **Speech Recognition Grammar Specification (SRGS)** is a W3C standard for how *speech recognition grammars* are specified.

2. **Grammar Specification Language (GSL)** is common speech recognition grammar format developed by Nuance Communications.
3. **Statistical Language Model techniques (SLM)**, whilst not standards in themselves, are another alternative to build speech recognition grammars. SLM approaches require the building of a statistical language model that assigns a probability to sequences of words. For example a 'tri-gram' SLM will look at the probability of three words in sequence. To achieve a good result, a very large number (usually 'tens of thousands') of utterances is required to produce a usable grammar. The SLM development cycle is time consuming and uses many resources.

Building natural language applications SRGS & GSL grammars

Many developers have only ever used an SLM approach to build a natural language grammar. This is because building and tuning an SRGS or GSL grammar to allow for natural language speech has typically been a very difficult task.

There is however an alternative to building SRGS and GSL grammars by hand. Inference Communications has developed a machine learning technique for building SRGS and GSL grammars using an approach known as '**grammatical inference**'. Essentially grammatical inference relies on using artificial intelligence to build your SRGS and GSL grammars automatically based on example utterances.

In the same way you build an SLM grammar using caller utterances, with grammatical inference, you feed the utterances into a '**grammar learner**' which outputs a set of grammar rules in whatever format you require (e.g. GSL or SRGS). The grammar learner has a fundamental knowledge of the language that you are building the grammar for (e.g. English) and combines this with your utterances to produce a set of grammar rules. Unlike an SLM approach, grammatical inference allows you to build a usable grammar with only a very small number of sample utterances ('tens of utterances' rather than 'tens of thousands'). Of course, if more training data is available, you can feed the grammar learner as much as you like.

With grammatical inference the building of a sophisticated multi-slot grammar such as an airline booking system takes about as much time as building a simple directed dialog grammar, this is because the complexity of the grammar design is handled by the artificial intelligence rather than the developer.

4 Grammatical Inference in Practice

Inference Communications has deployed dozens of applications that use the grammatical inference approach to building the application grammars. A typical example is the Inference 'Callback' system which allows callers to book an appointment in an online calendar. The grammars for this system are quite sophisticated allowing a caller to say things such as 'I'd like a callback no later than 8am Friday'. Building these grammars by hand would have taken a great deal of time. With grammatical inference the entire process is dramatically reduced.