The Design of Complex Telephony Applications Using Large Vocabulary Speech Technology

Steven J Whittaker
David J Attwater

BT Laboratories
Martlesham Heath
Ipswich
Suffolk IP5 7RE
United Kingdom

ABSTRACT

Almost every speech application involves integration with real world databases which may be large or complex. Telephony based examples include call-centre automation, customer identification and directory assistance. Many such applications are intrinsically large vocabulary problems with complex data requirements.

This paper illustrates the architectural, technical and dialogue issues relevant to the development of such systems. The implementation of the Brimstone corporate directory trial system developed at BT Laboratories is described.

1. INTRODUCTION

Complex speech applications typically involve the integration of separate component technologies such as recognition, synthesis and dialogue processing within an application architecture.

This must be linked to the databases associated with the particular business process. This application specific data can therefore play a key role within the dialogue. For example, a customer may be prompted for a number of items of information such as names, addresses or alpha-numeric references. These are often used to search the application database to retrieve additional information related to the task. Examples include directory applications where number information is obtained given spoken name and address information and transaction applications where an account holder is identified by name and reference information [1].

Typically, information from the application database is used on a vocabulary by vocabulary basis before or after pattern matching. However additional complexity can arise as such databases are often irregular and contain data of variable quality, having been designed with visual browsing or bulk mailing in mind and populated using a mixture of manual and semi-automatic methods. In addition, many forms of information such as names and addresses can be intrinsically variable - for example, multiple valid spellings of surnames, abbreviated first-names and complex geographical relationships. Processes are therefore required which allow the vocabulary of the application to be mapped to the vocabulary of the dialogue and which take into account the variety of ways in which the information may be presented by callers, and should be presented to them.

Just as the individual vocabularies of an application are typically constrained by the contents of the database, the contextual information intrinsic within the data can be used dynamically to maximise the overall application performance, improving both the accuracy and the identification and effective handling of responses outside of the domain of the application data.

The following sections discuss structures and processes which can be used to model the relationship between the contents of the application data-set and the operation of the speech processing components, and to allow the manipulation of multiple hypotheses within an application.

2. ARCHITECTURAL CONSIDERATIONS

Figure 1 shows a simplified architecture of a speech system which needs to access application data within the dialogue. The dialogue and information aspects of the system have been separated to ensure that dialogue design and related issues are not obscured by complex information processing.

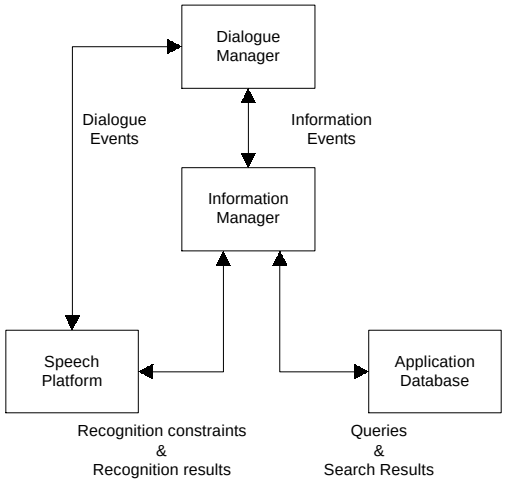

Figure 1 Large vocabulary architecture

The role of the information management component is to translate high level strategic information requests into tactical operations. These include interactions with the application database and the generation and manipulation of speech recognition vocabularies. Although, information management tends to be algorithmically complex, many forms of dialogue management can be well represented using graphical service creation tools. This allows the iterative design of the user interface which may involve end-users to be partially de-coupled from the underlying complexity.

3. DIALOGUE MANAGEMENT

As discussed above, many applications involve dialogues in which a caller is prompted for several items of information. The design of the dialogue will attempt to optimise usability and acceptability to provide an intuitive interface, whilst using strategies and messages are designed to obtain predictable responses from users which are within the constraints of the application and its database. A number of factors will affect such designs:

1. Callers are not likely to know all items with equal accuracy. For example, a caller is more likely to be able to give the name of their own road accurately in a customer identification application than they are the location of a business in a directory application. In some cases the items of information which might theoretically be most useful within the search may be the least likely to be known reliably, for example organisational codes.

2. Some items of information can validly be expressed in a number of ways. For example, many surnames can be spelt in different ways, many first-names have shortened forms and many geographic relationships can be complex - for example, an entry may be listed under a large town whilst the caller may be quote a smaller village or suburb.

3. The space occupied by a database is unlikely to be evenly occupied. For example, despite the fact that even in large data-sets many names are unique, there are 92 people named David or Dave Smith working within BT and 14 within the BT Laboratories.

These factors will affect the selection of those pieces of information for which the caller is prompted automatically, those which are requested only if necessary and those which are only ever used in confirmation. Whilst it is possible to ask the caller whether or not they know a particular piece of information, if used indiscriminately this can lead to unwieldy dialogues. Similarly, strategies for confirming and offering information must take into account the characteristics of the data-set.

Overall, the approach often adopted is to prompt, where possible, for predictable information and to do so in a way which provides responses of a predictable form. However, a significant amount of information processing may still be required and this is discussed below. Figure 2 shows a simplified dialogue model suitable for use in such circumstances.

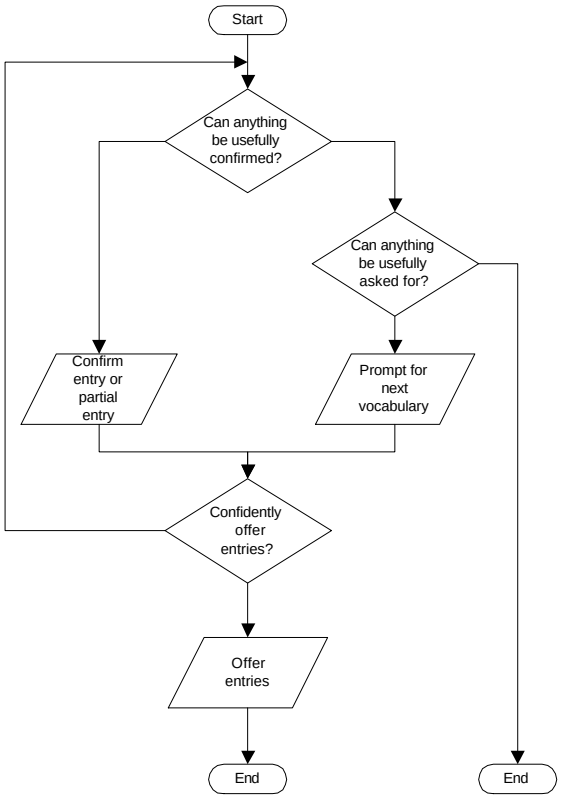

Figure 2 Simplified dialogue model

The model shows how vocabularies can be prompted in priority order. This order is based upon both usefulness, from an application perspective and usability. At every stage, the dialogue manager makes a decision as to whether to

- ask for additional information
- confirm all or part of the most likely entries
- offer a number of entries

During confirmation it is important to bear in mind the potential confusability of entries within the database. For example, a caller may draw no distinction between the names “David Bailey” and “Dave Bayley” and they must be managed accordingly.

4. INFORMATION MANAGEMENT

This section describes the general purpose information manager implemented for use in the Brimstone application.

The purpose of the information management component is to maintain database hypotheses based upon selected recognition results, independent from the surface structure of the dialogue. These hypotheses can be used to control the imposition of

application and contextual constraints onto the recognition process. During an interaction, the dialogue manager may interrogate the information model which can, for example, recommend whether the system is confident enough that confirmation may be attempted and that one or more entries can be offered.

The information manager is responsible for several distinct issues:

- the relationships within vocabularies (such as synonyms and homophones)
- the manipulation of database hypotheses (such as scoring of partial and complete entries)
- interactions with the application database

The key structure within the manager is the *data model*. This contains a number of *vocabulary models* each associated with one vocabulary within the application. These distinguish between items of vocabulary as represented in the database and the alternative ways in which they may be spoken and spelt by the caller.

Figure 3 shows a simplified version of a typical *vocabulary model*. Each arc of the model or combination of arcs can be represented as a *vocabulary map* which can be implemented using relational tables, file lookup or algorithmically depending upon the nature of the relationship to be modelled. In this way aliases such as synonyms or homophones can be handled in a framework providing tests for acoustic and semantic distinguishability.

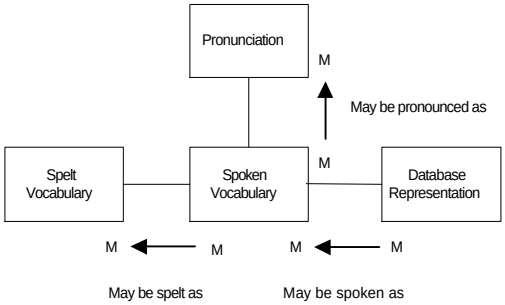

Figure 3 Simplified vocabulary model

The data model also manages the relationships between vocabularies as represented within the application data.

As information becomes available during the dialogue, hypotheses are generated which can be used to provide dynamic recognition constraints. It is useful to consider three ways in which the application could progress.

1. Selectively confirm candidates after recognition and re-prompt as required. Accuracy can be obtained at the cost of multiple repeats and a lengthened dialogue.
2. Collate the candidate lists associated with each recognition and use them together in a constrained database search. Database redundancy allows some recognition errors to be detected.
3. Narrowing the search after each recognition. Candidate lists are used to obtain contextual information from the database subsequently used to constrain further recognition cycles. Improved accuracy can be achieved but with poor rejection of errors.

In practise, these approaches can be combined to provide a more flexible approach to the manipulation of partial hypotheses.

Within this architecture, partial hypotheses are referred to as *tracks*. A *track* represents an ordered subset of the entries in the database based on application related information, the results of recognition and dialogue acts to date. The state of a given *track* can be used to adapt the recognition of subsequent vocabularies by applying revised statistical constraints, by dynamically creating new recognition vocabularies or by adapting other relevant characteristics of the recogniser.

Tracks can be straightforwardly manipulated and combined to perform complex processes. For example, if a dialogue prompts for a first-name and a surname, two *tracks* can be generated, one based on unconstrained surname recognition and the second on surname recognition constrained by the output of the first-name recognition. The resultant states can be compared to allow confidence based rejection and back off strategies. This process can be independent of the ordering of dialogue prompts.

Tracks and *vocabulary maps* are straight-forwardly extended to support vocabularies which are constructed from more than one database field, for example first-name/surname pairs, allowing major changes to the user interface with only minor changes to the underlying data model.

In addition, rather than relying on fixed dialogue order within the dialogue manager, it is possible to apply information theoretic approaches to dynamically determine the direction of the dialogue optimising the choice of subsequent vocabulary, or the values to be included in a partial vocabulary.

The information manager has been implemented as a flexible and re-usable object oriented system with a small application specific *assister* component as shown in figure 4.

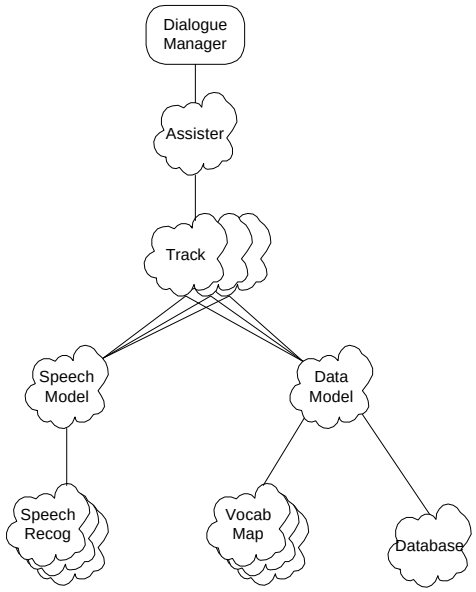

Figure 4 Simplified Class Diagram

Welcome to the BT corporate directory enquiry system.
Please say the surname.

Bailey

And now please spell the surname

B-A-I-L-E-Y

Please say the first-name

David

Is the name David Bailey?

Yes

I have three entries matching that information:
The number for Dave Bailey of Networks and Systems is ...
The number for David Bayley of Business Operations is ...
I'll repeat that ...

Figure 5 Example Brimstone dialogue

5. BRIMSTONE APPLICATION

A number of data intensive telephony applications have been developed using an implementation of the architecture described above. Brimstone is a prototype corporate directory enquiry system, which provides voice access to a subset of the BT internal telephone directory, and which is currently on trial within BT.

The multi-line trial system is based on the BT ISAP platform and the dialogue is implemented using the VISAGE graphical service creation tool.

Large vocabulary recognition and vocabulary creation are provided by the STAP recogniser and tool-set. The system makes use of the Laureate Text-To-Speech synthesiser.

During the trial, callers can obtain numbers for BT people in a geographical area which includes BT Laboratories at Martlesham Heath and nearby towns. This corresponds to approximately 5000 entries and results in vocabularies of about 3000 distinct surnames and 900 first-names. The database and the associated vocabularies are automatically updated on a regular basis.

The trial will provide detailed performance, usage and usability information across a broad motivated user base. Users also have access to a number of other methods for obtaining the information, such as on-line systems and a directory assistance service.

6. CONCLUSIONS

Within many speech applications, complex application specific database information plays a key role within the dialogue. This raises particular architectural and user interface issues.

This paper has described the design and implementation of an architecture which provides for the management of application information independently of the dialogue management component. This allows robust applications to be developed in which the information known by users can be used to in conjunction with inconsistent real-world databases.

7. REFERENCES

1. Whittaker S. J. and Attwater D. J., "Advanced speech applications - the integration of speech technology into complex services" *ESCA workshop on Spoken Dialogue Systems - Theory and Application*, pp 113-116, 1995

The following editions of the BT Technology Journal contain a range of background material and detailed papers on the speech platform, recognition, text-to-speech synthesis and service creation components described.

2. *BT Technology Journal - Theme: Speech Technology For Telecommunications*, Vol. 14 No.1, January 1996

3. *BT Technology Journal - Theme: ISAP and its application*, Vol. 14 No 2, April 1996