Experiences of Usability Evaluation of the IMAGINE Speech-Based Interaction System

R. I. Damper

Information: Signals, Images, Systems (ISIS) Research Group,

School of Electronics and Computer Science,

University of Southampton,

Southampton SO17 1BJ, UK

Email: `rid@ecs.soton.ac.uk`

K. Gladstone[1]

Deaf Education through Listening and Talking (DELTA),

3 Swan Court,

Peterborough,

PE7 8FD, UK

Email: `Keith.Gladstone@deafeducation.org.uk`

Abstract

IMAGINE is a recently completed multi-partner, multi-national European 5th Framework project aimed at developing systems for interaction with e-business applications through a multilingual natural language interface from mobile and other devices. The vision embodied in IMAGINE is universal access to electronic services by anyone (including disabled users) at any time. Since this long-term vision is not achievable with current capabilities in speech and natural language technologies, we have concentrated initially on a monolingual (English) application, namely a speech-enabled interface to an Online Shop. This paper is primarily concerned with usability evaluation of a prototype system for this application. The principal finding is that considerable challenges and barriers remain against the successful deployment of speech systems in real

[1]formerly Royal National Institute of the Blind.

applications. The problems are not just technical, but systemic (to do with the nature of speech) and are exacerbated by lack of shared background and difficulties of communication between different professionals involved in the design, commissioning and maintenance of speech systems.

1 Introduction

This paper is concerned with usability evaluation of prototype systems for interactive speech access to electronic business applications developed within the IMAGINE project, part of the European Union IST (Information Society Technologies) 5th Framework programme. The focus of IMAGINE was **I**nterfacing **M**obile **A**pplications with Voice Natural Lan**g**uage **In**t**e**ractivity. The project started from the technological basis established in a previous EU project (ESPIRIT 22252) called MELISSA, which supplied a capable natural language processing engine. IMAGINE has featured the participation of six European partners: Software AG España, IT Innovation (an autonomous, industry-facing part of the School of Electronics and Computer Science at the University of Southampton), Universität Saarlandes, Telefónica Investigación y Desarrollo, Royal National Institute of the Blind (RNIB) and Iniciativas Digital Media. The work reported in this paper was jointly conducted by RNIB and IT Innovation at the University of Southampton.

The vision embodied in IMAGINE is universal access to electronic services for everyone at any time with the ultimate objective of exploiting the full potential of electronic business-to-consumer services by providing a remote, multilingual, speech-enabled interface from a variety of mobile and other devices. These goals were deliberately ambitious to give the flexibility to explore the long-term potential of the technology, yet we also wished to deliver usable systems in the short term, as far as this is possible. Hence, to keep the research and development focused and we hoped achievable, the goals were initially constrained to providing systems for two separate monolingual e-business applications: one for an English site and the other for a Spanish site.

These were developed in parallel. This paper is concerned with the English language solution. The particular business application studied was RNIB's Online Shop—an on-line catalogue of products for blind and visually-impaired people (Gladstone, Dixon, Pick, and Alexander 2003). The specific interest in disabled users is consistent with IMAGINE's focus on universal access. Since visually-impaired people constitute a subset of general users but with some limitations, any system developed with them in mind should be usable by the wider public as well.

The remainder of this paper is structured as follows. In the next section, we briefly describe the main features of the prototype system developed by IT Innovation for speech-enabled access to the RNIB on-line catalogue. Section 3 then details the usability evaluation of the prototype. Section 4 briefly summarises our findings.

2 Design of IMAGINE Prototype

IMAGINE integrates the key technologies of speech recognition and synthesis, natural language processing, dialogue management, Web interfacing, etc. Every effort was made during the prototype design to follow the principles of good practice that are very slowly emerging in this area (e.g., Balentine and Morgan 2002; Cohen, Giangola, and Balogh 2004).

In line with the tenets of good system design and software engineering practice, IMAGINE is decomposed conceptually into three levels as depicted in Figure 1. The three levels correspond broadly to speech input/output, natural language dialog and the e-business application. If good separation can be achieved among the modules, with well defined communication between them, they can be separately designed, built and tested, so simplifying these tasks considerably. In the event, however, it was found to be extremely difficult to achieve good separation.

At the top level, the **mobile device and speech integration** modules are in charge of the direct interaction with the user. At the middle layer are those modules that

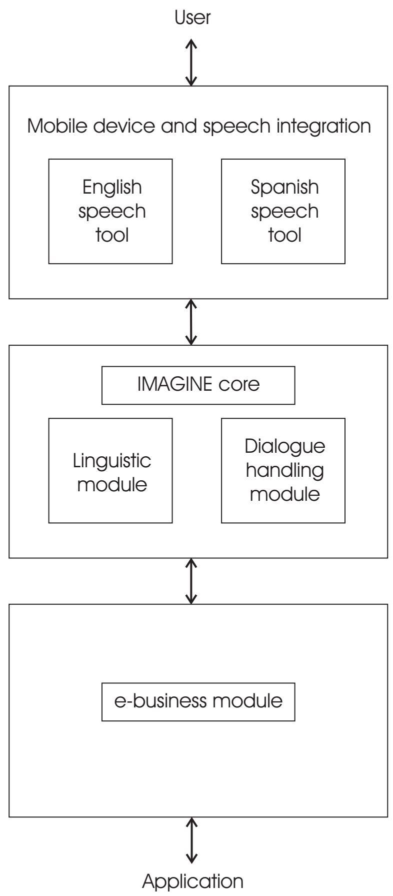

Figure 1: Three-layer structure of IMAGINE. The system is intended to be partitioned into separate modules handling speech input/output, natural language and the e-business application. However, this idealised complete separation has not been achieved.

control the natural language processing and dialog handling. The **linguistic module** is concerned with translating the user input, which has been converted from speech to text by the so-called English Speech Tool (EST), into an application-independent representation. (In this work, the Spanish speech tool developed by other partners is not used.) The **dialogue handling module** orchestrates the entire dialogue. Its task is to determine the precise nature of the user request, and to instantiate all the parameters related to the request. Together these two components constitute the **IMAGINE core** whose role is to interpret the user request and transform it into a form suitable for execution, including generation of output to the user (feedback,

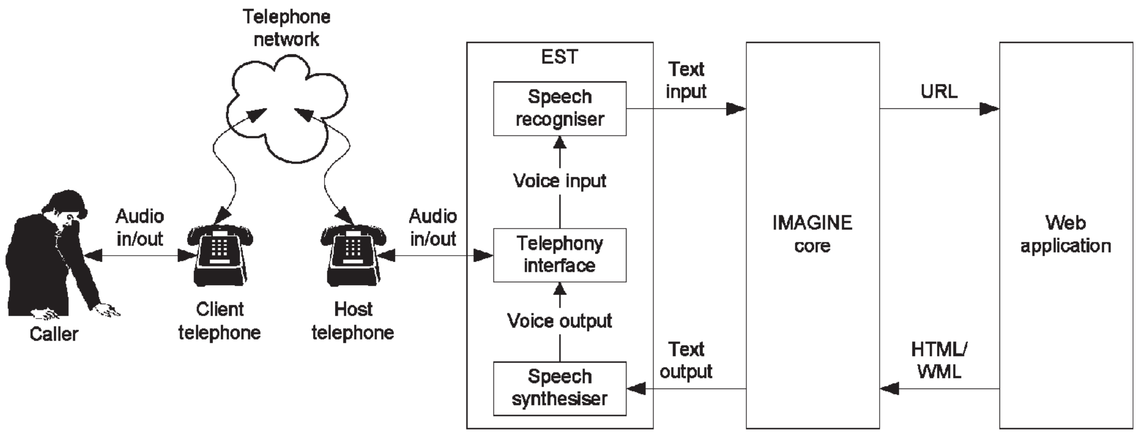

Figure 2: Role of the English speech tool (EST) in IMAGINE.

prompts, etc.). At the bottom level resides the **e-business module** that represents the specific application in terms of a general formalism. As stated above, the philosophy was to keep this module maximally separate from the rest of the system so that as little as possible, apart from the e-business module, had to be changed when migrating to a new application. However, but this proved very difficult to achieve in practice.

The EST is responsible for connecting IMAGINE to a telephone line, interpreting speech input and synthesising speech output. Its overall role in the system is depicted in Figure 2. (Note that the figure shows access by landline but this is as likely to be by mobile phone.) Again, the three-layer structure of IMAGINE is evident, with the EST residing in the top layer. The telephony interface connects the EST to an ordinary analogue telephone line via a Dialogic telephony card. When an incoming call is detected, the audio input is connected from the host telephone to the speech recogniser and the audio output is connected to the speech synthesiser.

The speech recogniser is responsible for converting words spoken by the caller into text. Since it was not part of the project to develop new speech and natural language technology but rather to study the usability of existing technology, we have used off-the-shelf products for this. The English language application is built on recognition (and synthesis) from Nuance (2002). This features a large-vocabulary, speaker-independent recognition engine purpose-designed for over-the-phone transactions, allowing (in

Nuance's words) "naturally flowing utterances from a wide variety of users". It also features as an integrated part of the architecture the Nuance Vocalizer text-to-speech synthesiser, used to generate user prompts and confirmations. The EST analyses the transcription provided by the recogniser and examines the confidence value associated with it. If this is below a threshold, the user is prompted to repeat or rephrase their last utterance. Otherwise, the input is processed by the recognition grammar (see below). The EST also determines if prompts or other speech outputs are appropriate to the user input and, if so, the necessary actions are carried out internally. A purpose-written prompt definition language allows randomly-selected, alternative prompts to add variety to the interaction.

Recognition grammars have been developed within the dialogue handling module to structure the interaction so that the recogniser 'understands' utterances relevant to the RNIB Online Shop application, e.g., "I want to place an order". The module applies basic interpretation to the input to filter out superfluous input (e.g., "I want to ... err ... place an order"), adds markup to complex types such as postcodes (e.g., "s o seventeen 1 b j" is converted to <postcode>so17 1bj</postcode>, and simplify complex input. For instance, recognition of proper names (which represents a real challenge to current technology) can be improved by entering them 'redundantly' both as whole words and as spelled-out letter strings, e.g., "Robert r o b e r t". The module converts this to "Robert". The implemented grammar was designed to interpret the following kinds of information:

- dates (day, month, year);
- letter names (facilitating spelled input);
- credit card details (digits, holder's name, expiry date);
- structured numbers (telephone, VAT, registration, etc.);
- titles (Mr., Mrs.), email addresses, passwords, sex, postcode, etc.;
- popular (top 95% in UK) first names and surnames;

- names, codes, categories and section names for products in RNIB catalogue;
- requests for login, product search, registration, change of stored details etc.;
- requests for help, changing speech output preferences, hang up, transfer call etc.

Various approaches to defining the recognition grammar were taken, initially comparing hand-crafted context-free rules and a statistical language model automatically derived from example text. The hand-crafted rules were found to be exceedingly difficult to write and to maintain as the size of the grammar grew. On the other hand, the statistical approach gave poor performance as a result of undertraining; a vast amount of example data was necessary for a large and complex grammar. Hence, a hybrid divide-and-conquer scheme was tried with a hand-crafted partitioning into modules (sub-grammars) according to function (e.g., registration, add product to basket, etc.) with each sub-grammar described by a statistical language model. The sub-grammars, being smaller, were more easily trained. This worked much better but, even so, the interaction between the sub-grammars, which had to be managed manually, became intractable with time. An automatic tool was written in an attempt to ease the problem, but was only marginally successful.

A key source of problems was that these approaches all attempted a *verbatim* transcription, rather than focusing on the *gist* of what was said. As a verbatim transcription is unbounded, this can generate an infinite set of outputs, making integration with other linguistic components difficult. The final method, as used in the deployed system, maps all possible input utterances to a finite set of outputs, by focusing on key words and phrases (e.g., extracting "place order" from "I want to place an order"). This was believed by the natural language processing partners on the IMAGINE project to provide a good solution, achieving better accuracy and speed of development, although it does have to be crafted by an expert grammar writer. It is at least possible, however, that this 'solution' has produced severe and unintended usability problems, as keyword spotting can lead to entirely erroneous interpretation of an utterance (e.g., "I don't want to place an order").

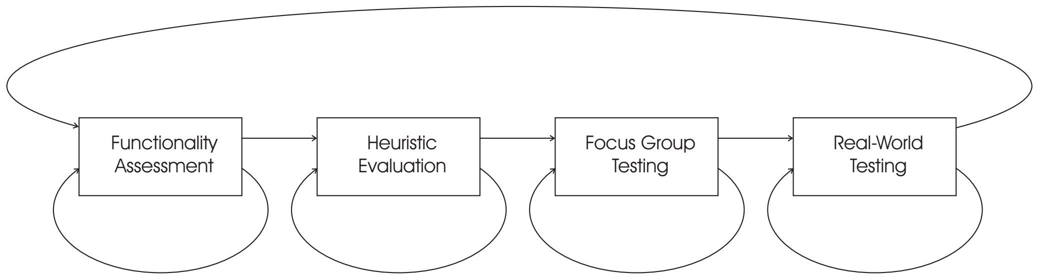

Figure 3: Schematic description of the intended scheme for usability assessment. The feedback arrows represent iterations of redesign based on the outcomes of each stage of assessment. In the event, only limited functionality assessment and heuristic evaluation were possible within the timescale of the IMAGINE project.

3 Usability Evaluation

The original validation plan allowed for extensive evaluations ranging from initial functionality assessment through heuristic evaluation and user focus group testing, culminating in garnering opinions of the target users in the envisaged real-world setting. Thus, it was intended that prototype refinement and improvement would be based on several iterations of evaluation followed by prototype redesign informed by that evaluation, as depicted schematically in Figure 3. However, time and other resource constraints severely limited our ability to put these intentions into practice. In particular, to avoid negative reactions and alienation of potential users, it was felt very important to avoid premature moves to the next stage of testing, and to do so only when the system performance was judged sufficient by RNIB's usability experts (see below) to offer a reasonable expectation of meeting users' needs. In the event, only basic functionality testing and heuristic evaluation were completed. At this point, and with user focus group testing still not deemed worthwhile, time for the project ran out.

3.1 Functionality Testing

Functionality testing is essentially unstructured, with the testers simply 'trying out' the system(s). Evaluators were informed of the functions implemented, and asked to test the systems to exercise these functions. At this stage, two evaluators tested the systems. For the IMAGINE prototype speech interface to RNIB's Online Shop, the tested functions included:

- log on, including initial registration;
- browsing without registration;
- finding products, via:
  - a name search
  - browsing through categories
  - an a-z search;
- putting products into a shopping basket;
- checkout, including entering/changing charging details and delivery address;
- miscellaneous commands (help, faster, slower, quieter, louder, repeat etc.)

In all, five different prototypes were evaluated for functionality, designated 1.0, 1.2, 1.3.3, 1.5 and 2.1. The intended functionality of each prototype is listed in Table 1. (Note that prototypes 1.0 and 1.2 appear to offer exactly the same functionality. The reason for 2 prototypes rather than 1 is that evaluators could not even log in with the first—see below—and this had to fixed before we could make progress.) In this numbering scheme, a leading '1' designates development systems that were intended solely for functionality and heuristic testing, whereas a leading '2' indicates that the system was (originally) intended for focus-group and real-world testing. Later versions were, of course, (re)designed to improve on earlier versions by taking account of the outcomes of the assessments as depicted in Fig. 3. Prototypes were released from the developers for functionality testing after validation, i.e., when they were deemed to be working to some useful degree. They were released (and evaluated) with default settings, prompts, etc.

	1.0	1.2	1.3.3	1.5	1.6	2.1	2.2
Log in/out	√	√	√	√	√	√	√
Change speech output preference	√	√	√	√	√	√	√
Ask for help			√	√	√	√	√
List product by letter			√	√	√	√	√
Browse catalogue by category/subcategory			√	√	√	√	√
Add product to basket				√	√	√	√
Show basket				√	√	√	√
List product by name/code					√	√	√
Remove product from basket						√	√
Check out						√	√
List product by approximate name							√
Register							√
Change registration details							√
Change delivery details							√

Table 1: Intended functionality of each prototype.

This validation in itself proved problematic, as there is a world of difference between a system developer checking out a speech recognition system on his/her own voice and the released system working satisfactorily with an independent speaker who has played no part in the development. The developer becomes unconsciously attuned over time to the vagaries of their system, and to the need for careful and consistent speech input. On the other hand, the RNIB evaluators were all (initially) novice users of speech recognition, although their expertise obviously increased over time. In the absence of relevant experience, novice users are likely to adopt the inappropriate model of speaking to another human when interacting with a machine by speech.

3.1.1 Prototype 1

In light of these remarks, it is perhaps unsurprising that neither evaluator could log on with prototype 1.0, although with further technical work (principally eradication of bugs) this was possible with system 1.2. In hindsight, prototype 1.0 was released too early (i.e., before the developers were happy with it) to satisfy the requirements of the project management plan, which specified what was to happen when. This kind of plan is, of course, essential to good management of a large and complex multi-partner project, and is always designed to cater for some degree of slippage. In this case, however, the technical difficulties of integration of speech technology into an existing software system, with minimal alteration of the latter, had been underestimated and the slippage exceed the maximum allowed for. To avoid having RNIB's usability experts left with nothing to do at a point in the project plan when they should have been fully active, the problematic first prototype was released prematurely (from a technical perspective).

At the point of evaluating prototype 1.2, spelled input of proper names, as listed in the requirements specification, had not been implemented. The usability assessors were aware that 'redundant', spelled input was planned by the developers with a view to improving recognition rates, but expressed the hope that it could be avoided. They felt that it would not enhance the ease of use of IMAGINE. At the current state of speech recognition technology, however, redundant, spelled input seems to offer more benefits than negatives.

Prototype 1.3.3, featuring the additional functionality listed in Table 1, was then evaluated and deemed unusable. Neither evaluator managed to complete a transaction. One commented "I ... had huge problems. It just could not understand me". The other, although unable to complete a transaction, still was able to offer the following comments and recommendations, listed in full to give a flavour of the sort of issues uncovered by the evaluation:

1. The number of rings before the system answers should be lessened.

2. "Would you like to browse the catalogue, order a product, register, or change your existing details?" should be added to the welcome prompt.
3. If the system does not understand, it should not ask for the same information indefinitely, but give the choice to be redirected to a real person.
4. The phrase "I'm lost" should be added to the dialogue.
5. When a user replies "no" ... the full welcome prompt should not be activated.
6. When the user asks to browse, all browse options should be presented.
7. When the user asks to search, all browse options should be presented, but without the deep-level `ListSubcat` function.
8. If the user says something like "list products by category", only the top-level category listing should be presented.
9. Category names and product names need better recognition so that the correct functionality can be provided to the user.
10. If prompted to state a product, the definite or indefinite article (e.g., "a kitchen timer") should be eliminated before search is carried out, to avoid erroneous search results.
11. It should be allowable to say "search" rather than "search criteria".
12. Only six products in a list are presented even when there are more than this available.
13. When a list of products is presented, the user hears "Would you like to order, hear product information or find another product?" If the user wants to "hear information", they should be asked to identify the product and not taken back a step in the dialogue.
14. Product codes need to be recognised when provided by the user.

It is noteworthy that some of these improvements are trivial to implement (e.g., 1 and 2) whereas others (e.g., 9 and 14) amount to complaints that the recognition accuracy is not perfect, which will be a fact of life for deployed speech systems for the foreseeable future. Yet others point to misunderstandings about how the underlying technology works (e.g., definite and indefinite articles are not used in the product search).

System 1.5, which was re-engineered to take account of those comments of the evaluators that were deemed practical to implement, was judged to be an improvement yet again neither evaluator managed to complete a transaction although enough experience of the system was gained to allow feedback to the developers. As before, there was a mixture of essentially trivial remarks (e.g., "the full welcome prompt should not be activated") and comments that allude to deep-seated aspects of system design and/or the state of the art in speech technology. For instance, one remark was "Pauses are too long" (reflecting the need for extensive processing of speech input and/or the realities of high network traffic on the Web) and another was "The system voice is not pleasant" (reflecting current shortcomings of speech synthesis that are not at all easy to overcome). Our experience over many years is that patterns of errors in speech recognition are very speaker-dependent (e.g., Damper, Lambourne, and Guy 1985; Damper 1988; Damper 1993; Damper, Tranchant, and Lewis 1996). However, this was not appreciated by the evaluators. There was a tendency for them to experience idiosyncratic problems with one or two specific words (e.g., "a problem with the system understanding category name 'hi-tech'", "It is apparent that the system does not understand the letter D") and to misinterpret this as a general problem that all speakers will encounter.

3.1.2 Prototype 2

At this stage, functionality testing moved to system 2.1, the first version of prototype 2. The purpose was to decide if it was appropriate to move on to heuristic evaluation with a larger panel of experts without further iterative refinement of the prototype. It was carried out over two days, representing approximately five hours total use of the system. The conclusion was that further updates were required if the subsequent evaluations were to be of value. The required modifications were fed back to the developers but, at this point, a virtual breakdown in communication occurred between the two sides. This breakdown is detailed not to apportion any kind of blame to either side, but because it illustrates a general problem in engineering complex but usable systems based on new technology.

The evaluators had noticed that if the house number and postcode entered by the user at checkout did not form a valid address (according to the postcode lookup software) the user was, quite correctly, asked to reenter their address details. However, IMAGINE required users to reenter *all* address details, not just property number and postcode. The evaluators quite reasonably asked for this to be changed, to avoid unnecessary reentry of already correctly-recognised parts of the user's address. However, the developers responded that this was not possible, since it required IMAGINE to understand what part of the address (consisting of multiple fields) was in error, and there were no automatic techniques to do this. The evaluators responded that the feedback to the user actually stated "no matching codes found" and so IMAGINE must either have pinpointed the error or the fedback message was in some way misleading. The developers in turn declined to answer this directly and reverted to stating "You are asking for the website to carry additional information about which field is at fault". They felt that this violated the philosophy that the e-business module should be application-independent. In the event, for whatever reason, be it different backgrounds and responsibilities, different assumptions about what is important, or something else, this problem was never resolved.

We are strongly inclined to believe that this, and several similar problems, had their origins in the physical (geographical) separation of developers and evaluators, which led to a lack of shared understanding and appreciation of each others' problems, constraints, competences and roles. One of the major conclusions of this paper is that, however hard it is to achieve, future projects of this type should not compartmentalise developers and usability experts in the way that we were forced by practical considerations to do. We are convinced that a much better result would have been obtained if the two sides had been able to work together in much closer proximity. As one of the usability experts wrote in an internal report: "This shows the difficulties in designing prompts when the technical issues are not overly clear. Prompts should have been reviewed with the developers to make sure that scenarios that are not possible at certain stages are not offered to users".

Again, this stage of the evaluation highlighted problems to do with current technological limitations (e.g., letter names being misrecognised in spelling mode). Further, some suggestions were made for usability improvements that could not be implemented because they implied redesign of the underlying e-business application. For example, it was agreed by all parties that a spoken numeric label to identify each product in a list would be helpful, but this is not part of the functionality provided by the Online Shop.

3.1.3 Summary of Functionality Testing

Following evaluation of prototype 2.1, a total of 49 problems were identified. These were grouped as follows:

- 16 problems were a consequence of speech recognition errors, mostly reflecting limitations of current technology (although good dialogue design can help).
- 11 problems were due to conflict between the way that user input is represented and processed in the underlying e-business application and in IMAGINE.
- 9 problems were to do with inappropriate prompts or wording.
- 4 problems resulted from missing product types.
- 3 problems were a result of errors in scripts used to analyse the textual response from the Online Shop.

- 2 problems resulted from some desired functionality not being present in the underlying application.
- 2 problems were caused by system design errors (e.g., months like "July" spoken as numbers like "seven").
- 1 problem resulted from a request by the evaluators for additional functionality that could not be implemented by the developers. (Because of limited knowledge of the application, it was not known if it could be implemented or not.)
- 1 problem (detailed above) was not resolved to the point of a shared understanding by usability experts and developers.

3.2 Heuristic Evaluation

Prototype 2.2 actually reached the stage of heuristic evaluation, as now described.

3.2.1 Recognition Accuracy

Before heuristic testing, prototype 2.2 was subjected to considerably more stringent, quantitative validation of recognition accuracy than were earlier prototypes. Reference speech data in the form of 3,000 words each spoken by 3 developers (2 English, 1 Chinese) were recorded on an Olympus DSS-330 Digital Voice Recorder. The speech data cover all parameter types (e.g., name, postcode) and functions (e.g., log in, add item to basket). The audio recording was then played into the recogniser. Such off-line validation is known to give a much more optimistic measure of accuracy than is on-line testing in which the speaker is concentrating on achieving a task. Word types were grouped (e.g., into number, postcode, product name, etc.).

The numbers of insertions, deletions and substitutions between the automatically recognised and reference transcriptions were determined and accuracy calculated as:

$$\text{accuracy} = 1 - \frac{\sum \text{insertions} + \sum \text{deletions} + \sum \text{substitutions}}{3000}$$

Initially, accuracy was assessed without dialogue in place (i.e., with the vocabulary artificially restricted to parameter types only, functions only, and letter names only). Parameter types were transcribed with an average accuracy of 93% whereas functions were transcribed with an average accuracy of 89%. However, letter names were recognised at just 76%.

Moving to a more realistic setting, a system-driven dialogue was imposed (i.e., the system determines what subset of the total vocabulary can be spoken at each point) as used in the IMAGINE prototype. The accuracy on a subset of 873 function requests was 68%. This avoids having to make all possible function requests available for recognition at one time, as would be the case in a user-driven dialogue. In this latter case, the accuracy fell further to 58%.

3.2.2 Prototype 2.2

The heuristic evaluations on prototype 2.2 were carried out at RNIB by five experts: two computer systems experts, two usability experts, and one blind computer programmer/accessible-technology expert. Of the two systems experts, one holds a degree in Computer Science and has around 5 years experience of developing systems for visually impaired people. The other (the second author of this paper) holds a first degree and PhD in Electronics and has been developing and evaluating systems for visually impaired people for 15 years. Of the two usability experts, one holds a first degree in Psychology and a masters in Ergonomics with some 5 years experience working on usability of systems with visually-impaired people. The other holds a degree in Linguistics and has been developing/evaluating systems for visually impaired people for about 3 years. The remaining evaluator is congenitally blind and has worked on developing and evaluating software for use by visually-impaired people for over 15 years.

Heuristic evaluation is deliberately more structured than the functional evaluation. The experts completed a score sheet designed to enforce consistency of response. It consisted of four columns: the first identified the part of the system referred to; the second was for a brief description of the usability problem; the third was to identify the usability principle violated; the final column was for recommended ways of ameliorating or eradicating the problem. In addition to a score sheet, evaluators were given a list of tasks to work through and technical notes on the functionality assessment to apprise them of known problems.

Subsequently, a consolidated table of identified problems was produced and one of RNIB's usability experts rated the severity of each using the (slightly amended) 5-point scale taken from Nielsen (1994):

1. don't agree that this a usability problem
2. cosmetic only
3. minor problem
4. major usability problem, important to fix
5. usability catastrophe, imperative to fix

All areas of operation were considered to exhibit usability problems (registration, log on, browsing, basket functions, checkout, etc.). Obviously, registration will be most potential users' first experience of the system, so it is important that this process is simple and quick. Yet not one evaluator managed to register at the first attempt. The shortest time to register was at the second attempt. Two people gave up after three attempts and two managed to register at the fourth attempt. A common experience was for the user apparently to complete the registration, only to be informed that the system had not responded at some point. The time to register was felt to be an unreasonable imposition on potential users. The system was found to have serious problems recognising names and numbers, as in addresses and birth dates. These problems were rated 'catastrophic' since they would have consequences such as preventing registration, credit card transactions being denied, and/or goods being delivered to the wrong address. On reflection, it was felt that many of these problems resulted from attempting to stick too closely to the format of the text-based visual interaction embodied in the original application. This was consistent with the philosophy of changing the underlying application as little as possible, but led to very wordy interactions and complex information interchanges as IMAGINE simply read out the relevant parts of the RNIB web application response.

In several cases, system and network delays caused a real usability problem in that the user was not sure that their speech input had been 'heard' by the system. Faced with a long pause, they were unsure of the cause and did not know what to do. This can, of course, also be a problem with conventional keyboard/mouse input to a computer, except that the immediate visual feedback when form-filling and the low error rates experienced with keyboard/mouse interfaces mean that users have far greater confidence that they have successfully entered data, and so are more likely to surmise correctly that network delays are to blame. When using a telephony interface, in which the potential to provide feedback on system status is limited and the user may be concerned about the cost of a call, pauses of uncertain origin and unpredictable length are likely to lead a user to 'give up' and terminate the transaction.

At some points, the dialogue was unduly constrictive and the system failed to 'speak the user's language'. For instance, the perfectly reasonable basket-function request to "change quantity" was not accepted and the user had to say (and had to *remember* to say) "update". Another thorny issue was confirmation by the system of the commands or data entered, whereby the system responds "I heard ......": is this correct?". Although this is a potentially important way of managing inevitable misrecognitions, it is also time consuming and tedious for the user to have to enter confirmations (themselves subject to misrecognition) especially when recognition is correct. This implies that confirmation should be used sparingly, but that itself introduces a problem because the interface then appears inconsistent to the user who is unable to tell in advance if confirmation will be required or not.

When checking out, it is obviously necessary to confirm the purchases a user wishes to make. However, the need to listen to the entire contents of the basket (and all the related product codes) is tedious and time-wasting, and represents a major usability problem. At heart, this reflects the unsuitability of speech, with the consequent need to remember the items on a spoken list, for this kind of interaction (Damper 1993). It was found to be extraordinarily hard, and unnatural, to speak one's name followed by its spelling (e.g., "John j o h n Smith s m i t h"). Users also found it very hard to remember and enter a credit card number of typically 16 digits without getting 'lost'.

4 Conclusions

This paper has reported experiences with the IMAGINE prototype system for speech interaction with an e-business application via the telephone line, focusing on the usability studies that were conducted as part of this large, multi-partner project. Severe usability problems were encountered. These were partly technical, but also systemic (to do with the nature of speech) and have uncovered both a lack of shared background and consequent difficulties of communication between different professionals involved in the project. This makes it inadvisable for the two to work independently with only infrequent meetings and/or discussion. Rather, for successful development of complex, interactive speech systems, a much tighter coupling is required with the two disciplines working and communicating much more closely than we were able to achieve.

References

Balentine, B. and D. P. Morgan (2002). *How to Build a Speech Recognition Application: A Style Guide for Telephony Dialogs* (Second ed.). Salem, OR: Enterprise Integration Group.

Cohen, M. H., J. P. Giangola, and J. Balogh (2004). *Voice User Interface Design.* Reading, MA: Addison-Wesley.

Damper, R. I. (1988). Practical experiences with speech data entry. In E. D. McGaw (Ed.), *Contemporary Ergonomics 1988*, London, UK, pp. 92–97. Taylor and

Francis.

Damper, R. I. (1993). Speech as an interface medium: How can it best be used? In C. Baber and J. Noyes (Eds.), *Interactive Speech Technology: Human Factors Issues in the Application of Speech Input/Output to Computers*, pp. 59–71. London: Taylor and Francis.

Damper, R. I., A. D. Lambourne, and D. P. Guy (1985). Speech input as an adjunct to keyboard entry in television subtitling. In B. Shackel (Ed.), *Human-Computer Interaction—INTERACT '84*, Amsterdam, The Netherlands, pp. 203–208. Elsevier (North-Holland).

Damper, R. I., M. A. Tranchant, and S. M. Lewis (1996). Speech versus keying in command and control: Effect of concurrent tasking. *International Journal of Human-Computer Studies 45(3)*, 337–348.

Gladstone, K., E. Dixon, R. Pick, and T. Alexander (2003). Approaches to voice interaction systems for use by visually impaired people. In *7th Conference of Association for the Advancement of Assistive Technology in Europe, AAATE 2003*, Dublin, Ireland.

Nielsen, J. (1994). Heuristic evaluation. In J. Nielsen and R. L. Mack (Eds.), *Usability Inspection Methods*, pp. 25–62. New York, NY: John Wiley.

Nuance (2002). *Introduction to the Nuance System: Nuance Speech Recognition System, Version 8.0*. Retrieved 8 March 2002 from `http://www.nuance.com`.