# A crowdsourcing approach to natural language automotive command interfaces

Jonathan Parr, MSc

Thesis submitted to the University of Nottingham
for the degree of Doctor of Philosophy

July 2017

# Abstract

Modern vehicles present drivers with a wide range of control options, the quantity of which have increased with technologies such as satellite navigation and advanced in-car entertainment systems. Conventional controls can cause safety-related effects such as requiring drivers to look away from the road. Voice interfaces offer the potential to reduce or eliminate visual distraction; however industry surveys show current implementations to be disliked by many users who consequently revert to conventional controls. This research considers why users reject voice interfaces and develops methods to address these concerns. Focus is given to the use of crowdsourcing methods to obtain geographically tagged command samples across a broad area. A simple token-based pattern-matching command processor is developed which bases command interpretation on command samples which are selected by geographic region and which therefore increase habitability by better matching each user's choice of command words. Crowdsourcing methods to enable user-selection of system response thresholds are also considered.

Study one elicited the views of 20 existing users of automotive voice interfaces. Analysis found that respondents did not find the pre-set command phrases intuitive, that a voice interface taking incorrect actions was particularly dissatisfying and that overall user satisfaction increased with the number of commands correctly recognised.

Studies two and three used 16 and 20 participants respectively to assess the potential to use crowdsourcing to build a corpus of typed modality commands and how typed and spoken modality commands might differ. In study two, command samples were terse and prompting using images of conventional controls resulted in ambiguity. Typed and spoken modality command samples contained both common and modality-specific words and phrases. The refined prompting method used in study three showed animations of actions occurring and this reduced ambiguity. The selected actions elicited more verbose commands and whilst individual participants often used differing phrases for spoken and typed modality commands, grouping multiple participants' commands significantly increased the proportion of common words.

Chapter five describes the development of a simple token-based command processing method which could make use of crowdsourced data to interpret voice commands. This method used a cross validation process to construct statistical models of expected sub-sentence pattern-matching for each action using a corpus of sample commands. Statistical models incorporated levels of expected pattern-matching for both intended actions and for non-intended actions. This method required only corpus data which could be obtained via crowdsourcing.

In study four, the model developed in chapter 5 was used to statistically evaluate the proportion of actions correctly identified for a set of spoken commands using spoken and typed samples

corpuses. Actions were correctly identified for 85.8% of commands using a spoken corpus and 87.1% of commands using a typed corpus. The percentage of cases where the intended action was one of the two actions with the highest pattern-matching level was 95.8% for the spoken corpus and 92.5% for the typed corpus which showed the potential benefit of user-disambiguation where pattern-matching levels are similar. Study five assessed the degradation in action identification in cases where a voice recognition method had failed to correctly recognise one word from a command. The percentage of actions correctly identified was not affected by the removal of command words for either modality.

The final study of the thesis comprised a crowdsourced user-trial of 55 participants who gave voice commands in response to animated prompts and then viewed responses from the command processor while adjusting response thresholds. Potential responses included taking an action without further confirmation, confirming an action before enacting it and asking the participant to rephrase the command. A disambiguation threshold allowed participants to set the degree of pattern matching similarity which would require a user decision. The study method required participants to compromise between different types of response error. Results showed that some participants preferred not to be asked for confirmation at any point and that once this group was excluded, it was possible to select a single set of thresholds which met the system-response choice of all participants.

This thesis shows that crowdsourcing is a suitable method to obtain command samples across a broad area to capture a range of command phrasing dialects and that selecting samples which match the speech patterns of specific users allows a simple pattern matching natural language voice interface method to robustly identify the actions intended by voice commands in a high proportion of cases. It was demonstrated that crowdsourcing can also be used to identify common system response levels to optimise user interfaces to best mitigate failures to identify an intended action. These findings propose a novel framework for automotive voice interface development with the potential to increase habitability and satisfy users across a broad geographic region and thereby realise the safety benefits of automotive voice interfaces.

Further research opportunities include the use of context-related data to improve command interpretation accuracy and the use of user-system interactions to further optimise voice interface responses. The sensitivity of the presented methods to a more diverse range of dialects may also be an area for study.

Opportunities for human factors research include social acceptability of using a natural language speech modality to control a vehicle and the opportunity to corroborate and expand upon the wide range of safety-related research concerning the potential benefits of voice interfaces.

# Acknowledgements

The research presented within this thesis was principally conducted at the University of Nottingham, between 2009 and 20015 and was funded by Jaguar Land Rover.

Working towards a PhD as a part-time student has been challenging to say the least and I would like to thank three elements which supported me along the way. Firstly, my thanks to the Research department initially and the Head of Technical Excellence latterly at Jaguar Land Rover for financial support over the protracted duration of the work. Secondly, I much appreciate the efforts of my supervisor Professor Gary Burnett in guiding me through the complexities of academic learning and methods. Thirdly, particular thanks go to my fiancé for patience whilst I dedicated much of my free time to the task and her efforts in proof-reading my thesis.

# Nomenclature

## Common terms

| | |
|---|---|
| Infotainment | A combined term for music systems including radio, cd, mp3 or other source and also video if provided. |
| Crowdsourcing | "Crowdsourcing is an online process that is distributed by the very nature of the Internet and it always involves the participation of the crowd. The rest of the characteristics depend on the proposed initiative." [1, p.10] |
| Voice interface | A human-machine interface which allows speech as an input modality. Speech may also be an output modality. |
| Voice recognition | The computational process of identifying words from spoken sound samples. |
| Habitability | The match between the language people employ when using a computer system and the language that a voice interface system can accept. |
| Full habitability | A voice interface system which allows users to employ any semantically meaningful, unambiguous and system-relevant phrase. |
| Semantic habitability | The degree to which a system can accept the meanings which users wish to express. |
| Syntactic habitability | The degree to which a system can accept users' choices of syntax or phrasing. |
| Insertion error | Non-language noise from user or ambient noise which a voice recognition method interprets as a spoken word |
| Substitution error | A voice recognition method identifying a different word to the one which was spoken. |
| Rejection error | The discarding of spoken words by a voice recognition method |
| Shallow or light parsing | Analysis of a sentence to identify the constituents (noun groups, verbs, verb groups, etc.). This does not specify their internal structure, nor their role in the sentence. |
| Vocabulary violations | A word given during word-based human-machine interaction which is not part of a computational vocabulary. |
| Command Processing | The identification of the value, intent or meaning of a set of words of a users choice. |

| | |
|---|---|
| Token | A set of characters (for example a word) assigned a discrete identity and in some cases a meaning. |
| Pattern-matching | Comparison of aspects of two or more items to identify a common pattern. |
| Artificial Intelligence (AI) | "The science and engineering of making intelligent machines, especially intelligent computer programmes. It is related to the similar task of using computers to understand human intelligence, but AI does not have to confine itself to methods that are biologically observable." [2, p.2] |
| Semantic constraint | The meanings which users are permitted to express in their inputs to a system. Typically, the application domain will restrict the scope of meaningful inputs. |
| Dialogue constraint | The meanings that users are permitted to express given the local dialogue context. |
| Syntactic constraint | The different paraphrases which are allowed for expressing a given meaning. |
| Lexical constraint | The individual words which are acceptable. |
| Voice command interface | A voice interface which allows users to specify an available action by speaking a word or phrase. |
| Fixed command interface | A voice interface which constrains users to one or more pre-determined phrase(s) for each available action. |
| Natural language command interface | A voice interface which allows a user to employ their own choice of command phrasing. |

## Definitions specific to this thesis

| | |
|---|---|
| Action: | A response which a voice command interface may enact in response to a suitably worded command. |
| Sample: | A sequence of words comprising a command given by a study participant for use as part of a command processing method. |
| Command: | A sequence of words comprising a command given by voice command interface user intending to enact an action. |
| Sub-sentence fragment | A pattern of words within a sentence. Examples include single words, two-words in sequence and two words with an undefined word between. |

# List of Contents

# List of Tables

# List of Figures

# *Chapter 1: Introduction*

## 1.1 Context

Computer hardware has for many years undergone a steady increase in processing power, approximately following Moore's law which proposes that processing capability will double every 18 months. Automotive systems have made use of this capability by providing users with increasingly sophisticated and complex features including music systems with large number of tracks and multiple play-lists, satellite navigation with live traffic updates and mobile telephone systems.

Increasing feature complexity has resulted in human factors challenges where methods must be found which enable a driver to control the additional capabilities. Physical control systems have been developed to perform multiple functions and touch-screen controls have been developed to allow many commands within a fixed space using for example, context based commands or command nesting. However, all of these methods have limitations including visual distraction where drivers need to look away from the road (as mentioned by Graham and Carter [3, p.155]), problems with quickly locating a required control and the achieving of a vehicle control layout which is both aesthetically pleasing and easy to operate, both noted by Burnett and Porter [4, p.523]

Voice interfaces have the potential to overcome or minimise these issues in many cases. A voice-interface that is fully habitable as defined by Baber [5, p.638] will allow users to employ any semantically valid and unambiguous command phrase, does not require memorising of commands and can therefore scale to manage future increases in feature complexity.

Voice interfaces generally involve four high-level operation stages as shown in Figure 1.1.

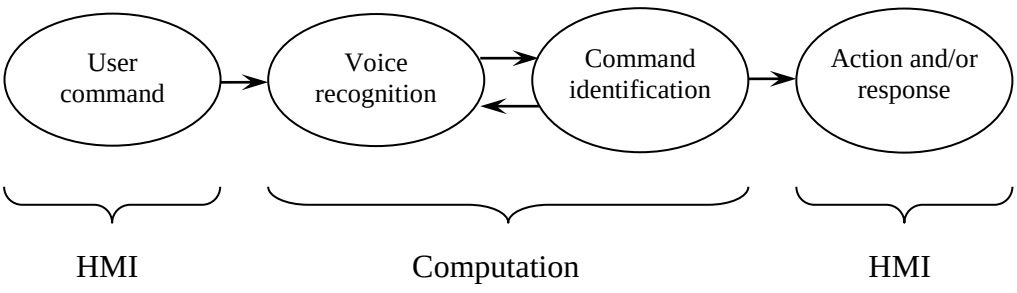



Figure 1.1. High-level Voice Interface operation stages.

The action identification stage can be broken down into four main types:

Fixed command phrase: Commands must be memorised. Some replacement words may be allowed but the grammatical structure is fixed.

| | |
|---|---|
| Key-word recognition: | Commands are tested for a specific words or sets of word which are then assigned an expected intent. |
| Decision-tree modelling: | Machine learning (primarily) is used to generate rules for phrase-patterns which are assigned to actions. |
| Statistical modelling: | Relationships between input factors are statistically weighted allowing confidence in a range of meanings to be considered. |

Commercial automotive systems investigated during this research were all of the fixed command phrase type. Study 1 detailed in chapter 3 showed that user satisfaction with these systems is generally low and that this is linked to low habitability and in particular, aspects of the fixed phrase method which is detailed in section 1.2.

By contrast, academic research is focused more towards statistical modelling and eventually artificial intelligence levels of voice interface where any utterance can be understood. Whilst this level of capability will resolve the problems with current systems, there is no clear point by which this might be achieved.

## 1.2 Problem statement

In 2012, an annual customer satisfaction survey run by an industry leading marketing information firm J.D.Powers found issues with voice command interface systems to be amongst those most commonly reported by users, particularly issues relating to operation of the audio, infotainment, and navigation systems. This was corroborated by study 1 (chapter 3). Responses given in this survey suggested that factors relating to the use of a low habitability fixed command method (i.e. allowing one or more pre-determined phrase for each available action) are contributory:

- Limited quantities of commands are recalled by users. This constrains voice interface scope and increases the likelihood of incorrectly recalled command phrases causing the voice interface to misunderstand commands.
- User frustration and cognitive load due to the need to manage incorrect actions or requests to repeat or rephrase a command.
- Increased cognitive load needed to recall the specific command phrase required by the voice interface
- Inability to invoke actions for which a command has not been previously memorised.

One way to alleviate these issues would be for automotive applications to implement one of the previously researched methods which would increase habitability by facilitating the use of natural language or more specifically, allowing any user to express a command using whatever semantically valid and unambiguous phrase they prefer. However, there are issues with each of these methods:

- Key word systems have demonstrated limited capability and scalability. Bobrow's "STUDENT" problem solving system [6] is an example of such a system which, whilst effective in specific circumstances, will identify meaning in gibberish if expected patterns are inadvertently recognised.
- For either decision-tree or statistical modelling to be useful in an applied automotive environment, the corpus on which the model is based would need to encompass the command phrasing which might be employed by users in this context. Issues relating to the applicability of a natural language corpus include:
    - Baber, Johnson and Cleaver [7, p. 56] note that the average length of utterance is reduced when users interact with speech recognition methods, demonstrating that the way people speak to computers is different to human-human and written language. If material on which machine-learning is based on human-human speech or on written material (not transcribed from a conversation), this material may not capture the command wording which users may employ in a human-computer context.
    - Users may express commands differently when under stress in an automotive context. Baber & Noyes [8] notes that "stress could alter the way in which operators respond to task demands, including requirements to produce speech."
    - Where a perception exists that a voice interface system has limited ability to understand a command, users simplify their language to try and assist in a variety of ways described in a range of literature summarised by Amalberti et. al. [9, p.4].
    - To ensure that a voice interface system will perform to an acceptable level of user satisfaction, people giving command examples may need to have confidence that the system is already at this level. That is, the ability of system to correctly interpret commands given by a new user expecting 'good' performance may be proportional to the confidence in capability of command identification method of people giving command samples for the corpus to be used for command identification.

The use of more advanced command identification methods may be assisted by the development of methods which facilitate the efficient gathering of data for a corpus which reflects the type of phrases which a user might employ in the desired context. Such methods would need to enable efficient obtaining of a number of examples of command phrasing which address factors including:

- Commands relate to a specific range of actions

- Includes any language and dialect within the region in which the method is to be employed
- The user is knowingly speaking to a computerised voice-interface system
- Commands are given within a perceived automotive context
- The user expects the system to understand the command.

Voice command interface performance for systems which are highly habitable and are able to accept natural language input can be measured by user satisfaction. Factors include:

- Whether the system is able to identify the intended action from a user's choice of command phrase, given a suitable information corpus
- The suitability of the corpus of information used for command processing
- Human factors considerations including action-identification confidence thresholds for types of response and the way in which the responses and user-interactions are managed.

In the case where the command interface corpus comprises sample command phrases, efficiency of command identification will affect the relationship between number of samples and user satisfaction. The relationship between sample-count and user acceptance will therefore be command identification method-specific.

The relationship between user acceptability, command identification method, response type, response threshold and sample count is complex and not currently clear. Figure 1.2 shows an outline of the relationship within the scope of this thesis.

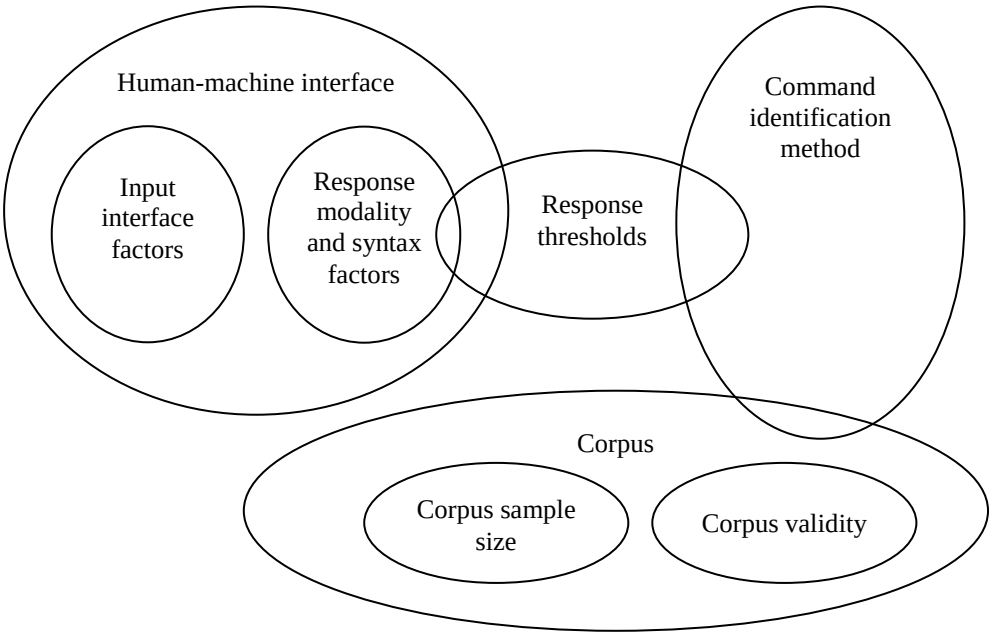



Figure 1.2. Venn diagram of factors which may affect user satisfaction with a Voice Interface.

One option to ascertain the required performance of a voice command interface system would be to simply ask a set of study participants what percentage of different types of system response they consider acceptable. It is however unclear the degree to which this will match actual acceptance in the context of live use of an automotive voice command identification system.

Minimally habitable fixed-command voice interface systems need only consider the confidence with which the voice recognition system was able to identify the spoken words and whether the spoken command phrase matches one of the pre-set command phrases. However, Furnas et al [10, p.964] note that (unless a user is able to memorise the phrases which the system requires), this can only be expected to cover a small number of user's choice of command phrase. Natural language command interface methods which attempt to provide full habitability (as defined by Baber, [5, p.638]) thereby allowing users to select any semantically valid and unambiguous command phrase must also consider confidence in correctly ascribing an intended action to the command phrase and a method of identifying a confidence level which is proportional to the probability of correct matching of a command to an action.

## 1.3 Response

Commercially available automotive voice interfaces assessed during this research used a fixed-command method; this type of method has low habitability and has demonstrated low levels of user satisfaction In addition to the potential for users to not know or recall the phrasing which the system requires, one factor which may affect this is the increase in command processing time due to the need for error correction as noted by Hone and Baber [11]. Whilst they allow alternative pre-set words and improve upon 'semantically adequate' systems which as defined by Watt [12, p.88] only allow one phrase for each action, they are nevertheless inflexible and cannot cover all command phrases a user may wish to employ. When artificial intelligence-based voice interfaces which approach human capability are realised, it is reasonable to expect improved levels of user satisfaction. However, Juang & Rabiner [13] describe the history of the technology and note that the development of such an interface remains a major challenge.

One option to improve habitability for automotive voice interfaces is to consider how command phrases which a system expects can be better mapped to the command phrases which users may choose. This could result in a (single or constrained variant of) fixed command phrase system which is better mapped to the preference of all users. Alternatively it may result in a natural language automotive interface which can accept any phrase a user may choose to employ. The first option seem unlikely to meet users' requirements as choice of command phrase may vary and the latter option may present difficulties if there are many languages and dialects in which users may express commands as well as variants in choice of command phrase. One option to provide a natural language automotive interface could be to subgroup the likely command phrases which the system can accept by the dialect and language

of any given user; in effect providing a language/dialect-specific natural language automotive interface.

In order to optimise a natural language automotive interface by likely choice of command phrase, users must be categorised according to the type of command phrasing they might employ and a way to do this is shown by Spruit [14] who notes that syntactic dialect is dependent on geographic location. Optimising command processing based on users' geographic origin required a method to obtain geographically tagged command samples from a wide region in a way which avoids the potentially impractical scaling of cost and time to obtain the required data. The first research question for this thesis therefore considered practical methods to obtain context-relevant, geographically tagged command samples from a large number of people over a wide geographical region. A number of methods might be used for this task, each of those considered presented challenges:

| | |
|---|---|
| 1. Simulator studies: | Simulator equipment is costly and may not be easily transportable. Collecting a large number of samples over a wide geographic area via this method may be impractical. |
| 2. Local surveys: | This requires people to perform the surveys and may therefore be unfeasibly expensive. It may also be difficult to ensure a common and relevant survey context. |
| 3. Remote home-computer sound-capture: | An attempt to achieve this is described in section 9.1.4 Capturing study participant's spoken commands was shown to present an information technology problem. |
| 4. Capture of data during live use: | Without an effective corpus, a method deployed to gather the data to generate one would firstly adversely affect users by performing poorly and secondly would only gather samples in the context of communication between a human and a poorly performing interface. |
| 5. Use of a different modality to enable crowd-sourcing of data | Data from a different modality may not match spoken commands. |

Simulator studies (option 1) and local surveys (option 2) both required resource which was proportional to the number of samples to be gathered. Since this quantity may be large, options 1 and 2 were discarded. Option 4 can only be considered when a natural-language command interface with a reasonable level of user-satisfaction is commercially available; since

this is not the case, option 4 was also discarded. The scope of research was therefore limited to options 3 and 5, both of which involve obtaining command samples via crowdsourcing.

To consider the performance of any sample-gathering method, the usefulness of samples in achieving a desired objective must be ascertained and this comprised a second research question. Methods by which this might be done include:

- Comparing samples to those gathered using a local spoken-command survey (from which samples are assumed to be of high value) using measures such as word count and noun/verb commonality.
- Considering performance of a command interface system for the objective performance metrics to which user satisfaction is most closely linked.

In order to measure the performance of samples obtained via methods researched in this thesis, a method of identifying an action intended by a command (i.e. a simple natural language command interface) would be required as a test facility and this comprised a third research goal. This command interpreter was required to:

- Operate based only on command samples and preclude the need for language-specific information (as command samples can be obtained by geographically broad crowdsourcing methods) by using simple pattern-matching techniques.
- Enable calculation of a statistically meaningful confidence level in identifying an action intended by a command phrase.
- Allow performance of samples captured via proposed methods to be directly compared to samples obtained using a local voice study.

It was not the intent to limit acceptable command phrases to the set of sample command phrases but instead to identify an intended action by considering lexical similarity of words or word sequences within the command phrase to the sample command phrases.

Factors which affect the performance of pattern-matching voice command interfaces may include how well the samples match commands given during a live usage context (sample quality) and the quantity of samples within the corpus. These two factors may interact. An additional factor is the habitability of a system using an optimal set of command samples given the efficiency of the command processing method; that is, the proportion of user commands which a system is able to correctly identify given an optimal sample-set. The relationship between quantity of samples, quality of samples and voice interface performance may also be affected by quality of voice recognition and the potential for command words to be misinterpreted. Investigating the effect of quantity of samples, potential interaction with sample quality and robustness to word recognition error comprises a fourth research question.

Ward et. al. [15, p.2] found that the degree to which voice interface systems take incorrect actions is a key measure for user satisfaction. Using this measure to consider sample quality

required that a threshold level between taking an action and other forms of voice interface response was first optimised. If this threshold were not established, a limiting case could require a voice interface to confirm every action with a user before taking it (which would make the measure ineffective whilst potentially reducing user satisfaction). Since there are additional response thresholds such as whether to ask for a command to be confirmed or ask for a command to be repeated or rephrased, there is a broader question of how to identify these thresholds, the degree to which fixed thresholds can satisfy a population of users and the effect of responses on users. Another consideration relating to thresholds is an ambiguity threshold where it is not possible to clearly distinguish between two or more actions. Investigating threshold levels is a fifth research goal.

## 1.4 Summary of research questions

Increasing user satisfaction with voice interfaces involves increasing habitability and maximising the number of commands which are correctly identified. Spruit [14] shows that the geographic origin of people speaking to an automotive command interface will affect the words used to express commands. One option to improve habitability without incurring a combinatorial explosion of command phrases as described by Baber [5, p.638] (and for a pattern-matching system, increasing occurrences of ambiguity between possible interpretations of a command phrase) is to align the information used to process spoken commands with the area from which a user originates.

The aim of this thesis is to consider whether crowdsourcing is an effective method of gathering information on which to base natural language command interpretations within an automotive context. This includes consideration of the effects of crowdsourcing modality, the method required to interpret commands using such crowdsourced data and the opportunity provided by geo-tagging of the gathered data.

### Research question 1: Method of gathering command sample data

To investigate the potential to develop a crowd-sourcing method for gathering geographically tagged, automotive speech interface command-phrase samples such that effect on choice of vocabulary is minimised.

### Research question 2: Validity of command sample data

To assess the degree to which choice of vocabulary varies between a spoken modality (which would be used to operate an automotive voice interface) and a typed modality (which facilitates crowdsourcing).

### Research question 3: Method of command processing using command sample data

To assess how effectively spoken commands may be interpreted using only crowdsourced example commands, focussing on a simple token-based command processing approach.

### Research question 4: Habitability of command processing method

To evaluate the degree of success and confidence with which the command interpretation method is able to identify actions intended by automotive-domain commands and the effect on success and confidence of action identification in cases where due to the voice recognition challenges present in an automotive domain, the command phrase has not been fully recognised.

### Research question 5: Method of gathering response threshold data and effect on command processing

To investigate the potential to develop a crowd-sourcing method of obtaining geographically tagged threshold levels for command interpretation system responses to optimally manage differing confidence levels when identifying an intended action.

# 1.5 Scope

## 1.5.1 Natural language automotive voice interfaces

Voice interfaces are currently in commercial use within an automotive domain. A range of benefits of using voice interfaces in place of physical or touch-screen controls within an automotive domain is noted in section 1.1 and related literature is described in section 2.3. The problem statement in section 1.2 highlights evidence that use of automotive voice interfaces is low and that therefore noted benefits are not being realised. For these reasons, the scope of this thesis is constrained initially to automotive voice interfaces and following the study completed in chapter 3, is further constrained to natural language automotive voice interfaces.

A fully habitable voice interface would, as described by Baber [5, p.638] allow the use of any semantically valid, unambiguous phrase to communicate any meaning or intent. McTear [16, p.1] notes that this unconstrained capability, requiring a complex world-view and artificial intelligence-based methods, has been a research objective for more than 30 years. An automotive voice interface is simplified by being limited to a constrained set of actions and this constraint was part of the scope of the command processing method developed in chapter 5. The scope was further simplified by constraining the set of possible commands to those which system users would employ; this is termed syntactic habitability by Baber [5, p.638]. For example, whilst a common vernacular such as ‘turn on the front headlights to dipped beam’ seems a reasonable candidate command, if it is a correct assumption that ‘send twelve volts of electricity to the front low-level road illumination units and thereby project light onto the external road surface’ would not be spoken in any real-word scenario, it would not be expected to provide a good match. Such ‘likely’ and ‘unlikely’ commands were not preset, but the concept that commands fell into these cases allowed the development of a command processing method which only considered the former. Figure 1.3 highlights the possible types of natural language automotive voice interface interaction within scope of this thesis.

## 1.5.2 Sample collection methods

Research on collection of sample command phrases was limited to crowdsourcing methods which did not require a researcher or survey person to be present. This was to avoid costs scaling to an impractical degree as the required number of samples increased and also to allow samples to be collected simultaneously (and therefore relatively quickly) from any country or region. Two such methods were considered:

- Use of computer equipment accessible to participants to collect spoken sound samples which are then translated using speech recognition techniques
- Use of computer equipment accessible to participants to collect typed modality data.

Issues relating to the first method were considered briefly, particularly issues related to the use of a speech modality which is not commonly in use in home computing. For the second method, the difference in command phrases between typed and spoken modalities was considered in terms of basic structure (word-count, proportion of words in both modalities), the probability of pattern matching a spoken command phrase and therefore assigning an intended action and ultimately, the effect on user satisfaction. The number of samples which were needed to correctly identify an intended action was also considered.

A number of factors relating to sample collection methods were excluded from the scope of this research:

- The degree of variation of command phrasing over geographical distances, population density and other factors. This would affect the number of samples needed across regions and the relevance of a sample to interpret a command from a particular user. It may have been possible for example to weight samples based on geographic distance from the location which best matches a user's dialect but this was not considered within scope.
- Live sample gathering. Once a system is functioning with acceptable user-satisfaction, samples could be gathered during live usage of the system with a similar effect to the user-specific optimisation proposed by Zudilova et al [17, unformatted]. This would be optimal in terms of context but may have privacy issues.
- Non-understood commands could potentially be informed by observing which conventional control was used following the voice interface being abandoned. This would however require a number of events to corroborate the link.
- The degree to which a driving-context affects choice of command phrase. Prompting via images of vehicle actions, computer screen animations and actual driving may result in variations in choice of command phrase.

### 1.5.3 Command processing method

The command processing method developed in chapter 5 compared commands to a set of sample commands which were previously given by people of a similar language and from a similar geographic location. The intent was not to achieve a direct match with one of these prior samples but instead to consider the similarity of the command phrase to sample phrases. For this reason, the developed command processing method was able to interpret command phrases which were similar in sequences of one or more words to those used by participants who provided the sample commands.

Three additional factors arose during this research but fell outside of the scope:

- Identification of the language and dialect of a user. To assign a subset of samples from many dialects to use as a corpus when pattern matching a command, it would be necessary

to identify the language and type of command phrasing employed by the user. This could be done by simply asking for the user's geographical origin (e.g. via touch-screen selection or voice interface dialogue), by sample search algorithm (although this may result in slow responses to initial commands) and may be enhanced by use of optimisation algorithms when the system is not in immediate use.

- The wording used by a voice interface in communicating responses to users is considered functionally in terms of disambiguation between actions which have similar confidence levels. The effect of differing response phrases are not however considered in this thesis. An interesting option would be to use research showing that users change their vocabulary to coincide to some degree with voice-interface responses (as suggested by Zoltan-Ford [18, p.545]) to develop an algorithm to optimise responses to match user command phrases. This would reverse the common idea of changing user input to meet system expectations, instead changing system output to meet user expectations.
- Watt [12] implies the concept of semantic habitability which in terms of a command system could be stated as the match between the actions requested by users and the actions which a system can enact. Communicating to users the set of possible actions was not considered.

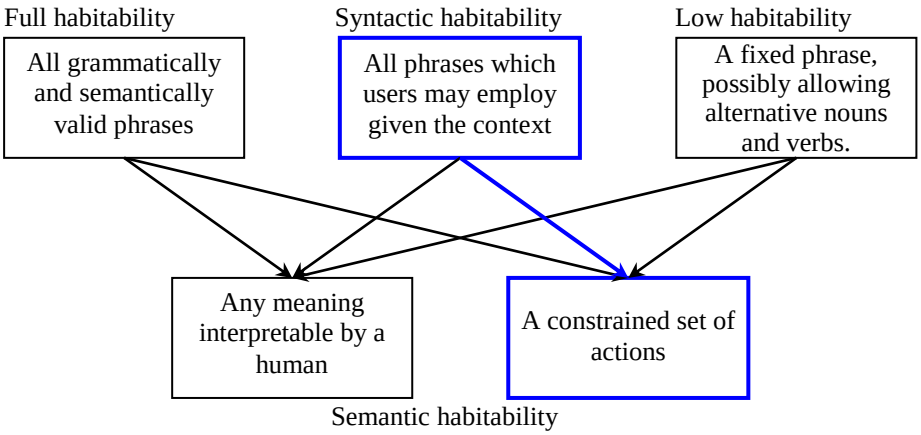



Figure 1.3. Scope for phrasing of commands and range of actions.

## 1.6 Thesis structure

### 1.6.1 Roadmap

| Chapter | Summary of content |
|---|---|
| 2. | A literature review was initially conducted to consider the broad voice-interface research area and also to look more specifically at the areas of crowdsourcing, command interface habitability and other human factors issues. |
| 3 | Study 1 (chapter 3): Considered subjective impressions of voice interfaces currently in commercial automotive use. |
| 4. | Study 2 (section 4.1): Assessed a method to obtain example command phrases from study participants by using a typed modality, crowdsourcing approach.<br>Study 3 (section 4.2): Extended study 2 by considering an automotive context and evaluating a developed, crowdsourcing prompting method and also increasing command length to better evaluate the proposed method. |
| 5. | This chapter describes the development of a simple empirical pattern-matching token-based command processing method which would be able to employ crowdsourced command examples to identify intended actions. The method was developed over four stages detailed in sections 5.3 to 5.6. This enabled the effect of differing modality on command processing performance to be considered. |
| 6. | Study 4: Evaluated information obtained from the studies in chapter 4 using the method developed in chapter 5 and considered the effect of differing command processor corpus modality and command modality on habitability and specifically command identification confidence. |
| 7. | Study 5: Measured how effectively the command processing method developed in chapter 5 was able to use crowdsourced command samples to robustly identify the action intended by a damaged command where one of the (two or more) words was not recognised by the voice interpretation system. The effect of using different modality corpuses was considered. |
| 8. | Study 6: Used a combination of data and methods from previous studies again using a crowdsourcing approach to present participants with a fully working command processing system and provide prompts for participants to give commands to enact actions. Participants were then asked to select response thresholds to optimise the resulting system responses. |
| 9. | Discussion and conclusions for the thesis. |

### 1.6.2 Synopsis

Chapter 1: Introduction

Chapter 2: Literature review

| | |
|---|---|
| Aim: | To identify valid research questions, it was first necessary to understand the research to up to this point and to ensure that limited performance of state of the art methods was due to gaps in knowledge rather than simply delay in realising already delivered research. |
| Findings: | Voice interface research is ongoing and has been broad and comprehensive. Questions remain relating to the quantity and context-related quality of context-specific command processor corpuses and how to elicit and use this information. |
| Conclusions: | Research detailed in this thesis was aligned to the questions identified. |

Chapter 3: User perception of current automotive voice-interfaces (study 1)

| | |
|---|---|
| Aim: | To assess user satisfaction with current automotive voice interfaces and factors contributing to views. |
| Findings: | People do not like the limited habitability, fixed command phrase systems and in particular dislike wrong actions being taken. |
| Conclusions: | Requirement to allow user-choice of command wording. |

Chapter 4: Exploring the potential for crowdsourcing command samples using a typed modality

Section 4.1: Preliminary user study on input modality (Study 2)

| | |
|---|---|
| Aim: | Research question 1 and initial consideration of research question 2. Typed command samples may differ from voice interface commands due to the modality difference. Also, the prompting mechanism used for crowdsourcing command samples may precondition study participants' choice of command phrase. The objectives were to identify differences in command wording between typed and spoken study modalities and also to observe the effects of the prompting mechanism. |
| Findings: | Statistically significant differences were detected between commands given by the different modalities. The prompting mechanism was shown to elicit a limited number of words and unexpectedly, moved focus from the intended subject to the prompting mechanism. |
| Conclusions: | Whilst differences in command phrases (between typed and spoken commands) were observed, these appeared to be mainly additional words. Further research is needed to understand the effects of these differences on command interpretation and if a problem, the degree to which this |

might be mitigated. The prompting mechanism needs to be further developed.

Section 4.2: Detailed user study on input modality (study 3)

Aim: Research question 1 and initial consideration of research question 2. The prompting mechanism used in study 2 may have affected the differenced between commands given in the two modalities, the length of commands. The objective of this study was to evaluate an enhanced, crowdsourcing prompting mechanism and review the effect of this on modality effects. In addition, an objective was to obtain a set of command samples suitable for the development of a command processing test tool.

Findings: Use of animations showing actions occurring generated correctly focused prompts. The resulting commands were longer. Modality differences remained.

Conclusions: Resolving the issues with the crowdsourcing prompting mechanism resulted in a data-set which still shows clear modality effects. It was necessary to develop a command processing test method to evaluate the significance of these differences.

Chapter 5: Methods for deriving command processing confidence levels

Aim: Research question 3. To understand whether simple pattern matching of a command to a corpus of crowdsourced command samples will give a high degree of correct action identification. To derive a confidence level on which a decision on the type of response to the user might be based. Investigation of methods for managing ambiguity where commands may mean two or more actions due to inherent ambiguity or due to unspecific input.

Findings: Simple statistical pattern matching is very capable at identifying correct actions under the stated conditions. Confidence levels can be identified by using recursive corpus testing and ambiguity can thereby be quantified.

Conclusions: The test method both works unexpectedly well and also supports the principle of basing a command processing method on command samples recruited via crowdsourcing from people whose choice of command phrasing is similar to that of a user.

Chapter 6: Confidence level and probability of distinguishing between actions (study 4)

Aim: Research question 4.

| | |
|---|---|
| | The degree to which modality effects on command wording can now be considered from an end-effect perspective by using the command processing method developed in chapter 5 to compare confidence in identifying intended actions. |
| Findings: | Modality differences have a measurable effect but it is very small. |
| Conclusions: | A crowdsourced, typed modality sample-set (corpus) is a viable basis for spoken modality command processing. |

Chapter 7: Robustness of command identification to voice recognition errors (study 5)

| | |
|---|---|
| Aim: | Research question 4.<br>To assess the robustness of spoken and typed modality corpus-based command processing to voice recognition errors resulting in one non-recognised word in a command. |
| Findings: | Reduction in command identification for commands as given and commands with words randomly omitted (to simulate voice recognition issues) is minimal and not significantly affected by corpus modality. |
| Conclusions: | Typed samples collected via crowdsourcing using the prompting and interaction methods described in chapter 4 provide a good basis for interpreting spoken commands where not all words have been correctly recognised. |

Chapter 8: Selecting optimal system responses by confidence-level thresholds (study 6)

| | |
|---|---|
| Aim: | Research question 5.<br>To evaluate the degree to which command processing system response to confidence levels (in correctly identifying an intended action) affects user satisfaction. The objective is to have participants of a crowdsourcing study identify their choice of response thresholds and to consider the possibility of thresholds which are universally acceptable. |
| Findings: | Whilst threshold data does not form a good parametric distributions, results show discrete ranges and fixed thresholds can be identified to generate optimal responses for all study participants. |
| Conclusions: | It is likely that the optimal response levels identified for the study participants are suitable for all people of a similar type to those in the participant set. Whether these or other levels are suited for other sets of people is an opportunity for further study. |

Chapter 9: Conclusions and discussion

### 1.6.3 Outline

In the 1920's, a toy called 'Radio Rex' was able to respond a person speaking its name (based on detecting 500 Hz frequency energy). From this very basic start point, voice interfaces have grown gradually more sophisticated and are now able to respond to many (correctly worded) commands. However, despite the development which has taken place, a 2012 J.D.Powers study has shown that in an automotive context, public perception of voice interfaces remains poor. There are also examples of humour surrounding voice interface usage in popular culture, generally noting the fruitless efforts of a user attempting to communicate with such a device.

A wide range of research cited within this thesis has found that, within an automotive context, there are many potential safety-related benefits of employing a voice interface. For example, drivers need not look away from the road when using the controls and there is no increased difficulty and eyes-off-road time when learning a control interface on a new car. Nor is there a need for designers to find acceptable conventional control layouts to accommodate the increasing functionality present in modern vehicles such as satellite navigation and entertainment systems.

Where user perception of voice interfaces is poor, users may elect to employ conventional controls even when voice controls are available, thereby eliminating the potential advantage of voice interfaces. Overcoming poor user perception of voice interfaces requires consideration of factors which may cause such perception. These may include external information such as press reports of poor capability (possibly based on earlier, less capable systems), low proportions of commands identified (including on systems where a period of usage or training is needed to optimise a system), poor voice recognition, low habitability resulting in the need for users to memorise commands (and low likelihood of command recognition should they fail to recall correctly) and poor human-machine interaction (both generally and specifically in cases where commands are not recognised).

The semi-structured interview described in chapter 3.1 addresses the question of relative importance of factors, finding that a key factor in user perception of voice interfaces was the proportions of commands correctly identified and that this was affected by the low habitability of the voice interface and the consequent need for users to memorise and state specific commands. The question then arose as to why natural language voice interfaces with higher habitability and which consequently permit the use of any command phrase were not in common automotive use.

One challenge for automotive voice interfaces is that a vehicle may be sold in many areas of the world and that the voice interface will therefore need to work with many languages. Also, in terms of interpreting a natural language command into the desired action, that there are

many dialects means that there may also be many ways of phrasing a command within the same language. If it were desired to provide a comparable capability to all users given any language or dialect, this is easily done by providing fixed commands which users must learn. However, in the case of natural language interfaces, capability may depend on two factors; firstly the type of natural language processing and secondly, the available information (corpus) used to identify the meaning of a command.

In the case of key word methods, commands would be recognised based on rules of whether words are included in or excluded from a command and this requires understanding of what commands people are likely to say (which is dependent on language and region). Similar information is needed in the case of more sophisticated pattern-matching methods such as that developed in chapter 5. In the case of more complicated semantic-understanding methods, command identification may be based on modelling large corpuses of language. However, material cited in this thesis has noted that human to human speech is dissimilar to human-computer speech and that the context in which the speech is given may be important.

It was noted from the above that methods may be required which facilitate the gathering of large numbers of commands given in an automotive context from a wide geographic region without incurring impractical scaling of resource demands. An internet-based crowdsourcing method was considered and this raised the question of whether the resulting typed-modality command samples might result in different command phrases to those given in a spoken modality (such as when a voice-interface is in use). This concluded that the modalities did result in command phrase differences.

A further question was the degree to which this measured difference would affect proportion of commands correctly identified and it was noted that evaluating this would require a command processing method which could identify actions using only crowdsourced command examples as an information basis. To address this question, a simple token-based pattern-matching method was developed. Assessing spoken commands against both a spoken and typed corpus of information using this method found that there was no statistically significant difference in command recognition. It was also found that the percentage of commands for which the correct action was identified was 87.1%, close to the approximately 90% threshold suggested by participants interviewed in study 1.

The potential to use a simple pattern-matching command processing method given a crowdsourced corpus of command similar in dialect to a user gave rise to human-factors questions. Firstly, based on a calculated degree of confidence in identifying the action intended by a command, whether a common response threshold (e.g. selecting between enacting a command and confirming a command) might be acceptable for all users. Studies performed demonstrated that for the set of study participants, if users were first asked to

identify whether they would prefer to confirm commands (where necessary) or if in all cases they would prefer to be asked to repeat a command, it was then possible to identify response thresholds which were common for all participants.

Robustness of the proposed methods was also considered, specifically the question of how the proposed methods might fair when the words of which a command phrase comprised were not all recognised by the voice recognition system. A study assessing commands with a word omitted found that using the proposed methods and for the commands assessed, there was no statistical difference in probability of command identification where one word was omitted from commands given.

# *Chapter 2: Literature review*

## 2.1 Introduction

Aspects of voice interfaces relate to a broad span of academic disciplines. One area of research comprises engineering and human factors, another relates to computational science and another relates to psychology and linguistics. This thesis examines the use of automotive natural language voice-command interfaces and this review is therefore constrained to the specific areas relating to this remit.

This literature review will consider early research to provide a developmental context for the more detailed topics and note how academic research has progressed. Literature relating to the effect on vehicle safety of interface design and operation is then considered with a particular focus on the eyes-off-road consequences of conventional interfaces as this provides one of the key drivers for the development of voice interfaces. The potential for voice interfaces to engender cognitive load is also considered as this needs to be carefully considered in any method development. Since the various types of control system vie for space within reach of a user, aesthetic effects are also considered.

A key area of research examined is the identification of an action intended by a natural language command. Implementing a fully habitable system which allows users to instruct a voice interface system using whatever command phrasing they wish may offer the least cognitive burden, require the least user preparation and eliminates the possibility of error due to the use of incorrect command phrases. There are several methods of natural language processing and these are reviewed. One challenge is in obtaining the information which is required to identify the intent of spoken command. Such information may need to be robust to different types of command phrasing and research considered addresses regional dialects. Since crowdsourcing is a possible method for obtaining the information needed to engender effective command processing, research on the possibility of using a different modality to facilitate this is also considered. In order to develop methods relating to voice interfaces, a process for studying and/or measuring such methods would be needed and research relating to the Wizard of Oz method is assessed.

The concept of habitability (i.e. the alignment between the language people use when speaking to a voice interface and the language that the interface can make use of) is considered. Some factors affecting habitability are examined as well as some consequences of low habitability. Different categories of habitability (e.g. syntactic, semantic) are noted.

The final part of this literature review relates to interface design. One area of research considered relate to users' choice of command phrase, what might affect this and whether it can be constrained to simplify the task. The other interface topic considered is how best an

interface might manage a poorly understood command so that user satisfaction is least affected by the recovery process.

## 2.2 Inception of voice interfaces.

### 2.2.1 Early voice interfaces in fiction

The concept of using of speech for human-machine interaction has been around for a considerable time. An early fictional use of human-machine voice interfaces was described in the 1920 play R.U.R by Karel Capek. More contemporary speculative examples include Asimov's 'Multivac' computer and the ship computer from the Star Trek franchise. These recent fictional interfaces were able to understand and answer a question on any topic using a natural language interaction. Whilst such interfaces offered the opportunity for authors to explore a range of philosophical artificial intelligence issues, flawless operation of voice interfaces was assumed, both in recognising human commands and the machine responses (communication or action). Whether such systems were envisaged for similar reasons that voice interfaces are now desired or simply to provide literary tools (such as exposition or a morally unambiguous antagonist) are not clear.

### 2.2.2 Initial research

The areas of research from which a voice interface might be constructed were developed at different times. Speech synthesis was one of the earliest components realised; the VODOR (Voice Operation DEmonstratoR) was built by Bell Laboratories in 1936 as described by Dudley, Riesz and Watkins [19]. This required keyboard input to generate electronic analogues for the sound of syllables being spoken. The method demonstrated that machines could produce human language speech sounds but incorporated no computational structure for these to be devised from words or sentences.

An early example of speech recognition was the "Audrey" system developed by Bell Laboratories in 1952. Davis et. al. [20, 639] note that this was constrained to only recognize the digits 0-9 spoken by a male voice and that Audrey was accurate to 97 to 99 percent as long as the machine was adjusted for a particular user and the user paused 350 milliseconds between words. Constraints on the number of words which could be recognised were largely removed by the development of the Hidden Markov Model which enabled a probability for an unknown state to be predicted from a series of stochastic factors. The mathematics underpinning the Hidden Markov Model was developed by Leonard E. Baum and team from 1961 to 1972; papers relating to this include “A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains” [21].

One option to identify an action or response required by a spoken statement is to simply link a specific word or word sequence to a pre-set response. Early development of methods to improve on this by allowing use of natural language began with simple key-word systems

which assigned intent by identifying specific words in spoken phrases. G. W. Furnas et. al. [22, p.1753] note that random pairs of people use the same word for an object only 10-20 percent of the time. They point out that capturing a range of words for each noun is one option to improve this result.

## 2.3 Basis for Voice Interface research

### 2.3.1 Safety considerations for conventional interfaces

A key motivation for the development of voice interfaces within an automotive context relates to safety. Operation of simple button controls can require a glance to locate the control, other feedback-based controls such as touch screens may require a longer duration for action-response visual interactions. Graham and Carter [3, p.155] noted that "speech modality is both hands-free and eyes-free, meaning that the hands can be kept on the wheel and the eyes on the road."

Several studies have been performed to evaluate the effect of visual distraction and/or cognitive load on road safety. Greenberg et. al. [23, unformatted] reviews a range of these studies from authors including Parkes and Hooijmeijer [24] and Strayer et. al. [25] and notes that whilst they evaluate performance change against low-level metrics such as missing traffic signals, it was 'difficult to assess the magnitude of the on-road problem from these studies'. They then used an articulated driving simulator to evaluate a range of vehicle controls including hand-held and hands-free phones and concluded that different control tasks affected different aspects of driving performance:

- Using a single rocker switch resulted in brief and quickly corrected variation in driving direction.
- Adjusting the vehicle climate system using three controls had a significant (negative) impact on both driving direction and detection of events behind the vehicle.
- Using a mobile telephone handset to dial a number and to retrieve and respond to voicemail resulted in significant lane violation errors.
- When using a hands-free mobile telephone to dial a number and to retrieve and respond to voicemail, there were no significant effects for any driving metric measured.

The findings of Greenberg et. al. show that different types of conventional controls negatively impact driving in different ways and also show that in two cases, use of hands-free voice control mitigates this effect. It does not however identify whether the benefit of hands-free controls were due to removing the visual distraction or some other control modality effect. One additional result identified by Greenburg et. al. [23, unformatted] related to the answering of a mobile telephone call using either the handset or a hands-free activation button located on the rear-view mirror and in both cases, asking the caller to call back later. In both cases, there

was a similar (significant) reduction in detection of events happening in front of the vehicle. However, it was noted that uniquely in the case of the hands-free system, drivers were noted to be looking through the windscreen while still not detecting the event. This was not the case where the handset was used. One possibility is that the handset provided a visual and perhaps object-handling distraction but that when this was removed, a masked cognitive load effect resulted in a similar reduction in driving performance. Whilst the safety effect (in an automotive context) of visual tasks may be measured and controlled by action or response time irrespective of the reason for the distraction, results showed that in some cases and for specific measures, voice-interfaces can mitigate the degradation in driving quality caused by operating conventional interfaces. In other cases, reduction in driving quality is similar when using voice-interfaces and that this needs to be considered during interface design.

A large range of control options may require a similarly large space for conventional controls and Graham and Carter [3, p.155] state that "The "dashboard real estate" available for controls and displays is very limited. Traditional controls, such as keys and buttons, must therefore be used sparingly". One effect of this limitation combined with increasing vehicle technology such as satellite navigation may be an increase in dynamic control methods such as touch screens. Burnett and Porter [4, p.523] agree with this finding, noting that "for many cars, it can be problematic to find the required control quickly, carry out the appropriate operation and be confident that the correct action has been made".

Safety issues with conventional interfaces specific to vehicle navigation systems were assessed by Burnett et. al. [26] who collated and considered a range of material relating to driver interaction during destination entry while driving. They concluded that there are 'relatively high levels of distraction that can result from carrying out common input tasks with current designs for in-vehicle navigation systems' and that legislating to disallow such interaction 'inhibits the potential for novel design solutions'. They also note the use of speech modality as one such potential solution.

### 2.3.2 Voice Interface considerations

Experiments performed by Lee et. al. [27, p.9] show that "speech-based interaction with in-vehicle information systems places a cognitive load on drivers that can affect driving performances. Speech-based interaction draws upon some of the same cognitive resources as driving and so can distract drivers just as visual displays and manual controls can". The specific case tested in this study involved hearing and responding to emails which required some mental focus. It is possible that the measured effect may have related mainly to the cognitive task and not the interface modality and that the modality merely enabled a task of this nature to be performed while driving. This was supported by Bigot et. al. [28, p.1050] who performed a study comparing the four combinations of spoken or typed input and spoken

and typed responses, noting that "mental workload was found to be greater in our study when a speech component was used to perform a relatively complex task."

Rather than the conclusion being to consider voice interfaces as a basis for cognitive load and therefore a potential safety concern, it may instead be the case that care needs to be take when either implementing a voice interface which has inherent complexity (e.g. a requirement to recall interface nested menu structure) or when considering additional cognitively demanding tasks facilitated by the modality.

The effect of varying command recognition accuracy on driving performance was noted by Kun et. al. [29, unformatted]. They identified a significant correlation between recognition accuracy and steering wheel angle variance where lower recognition accuracy resulted in higher steering input (to maintain a similar accuracy of lane position). They conclude that this indicated “worse driving performance". This indicates that to achieve a required degree of driving performance may require a voice interface performance above a threshold level. However the precise relationship between voice interface accuracy and driving performance and therefore the threshold of voice interface accuracy required is not known.

### 2.3.3 Aesthetic considerations

One additional reason why voice interfaces may be beneficial is in providing the user with an aesthetically pleasing and uncluttered environment. Gellatly [30, p.2] considers the potential for speech modality to facilitate this, explaining that "Speech recognition technology could reduce the number of hand-operated controls ... reduce clutter inside the vehicle ... reduced response complexity, which in turn may reduce driver workload". The reduction in clutter would only be fully realised where a voice interface replaced (rather than supplemented, as is commonly the case) conventional controls. Bradford [31, p.63] states that “Work needs to be done to identify situations in which speech is the preferred medium of interaction” however this may relate to user experience of voice interfaces which may depend on how often speech control correctly interprets commands as noted by Kun et. al. [29, unformatted]. As voice interfaces are improved, a state might be achieved where voice control became the preferred modality and in this case, it could be possible for conventional controls to be abstracted in screen menu layers or made less prominent. The proposal that voice interfaces reduce control complexity could be better stated as a reduction in visible and/or primary control complexity as navigating a virtual (and non-visible) menu structure could (depending on interface design) prove as complex as visually navigating a physical control-set.

## 2.4 Knowledge-basis for command processing

Command processing methods can be based on grammatical structures which identify likelihood of word sequences by probabilistic rules. These can be constructed from material which was written for the consumption of humans or from records of human-human speech. Bradford [31, p.63] notes that this context may not result in the correct syntax for a speech command interface, stating that “Human/computer communication generally consists of one syntactically complete utterance (roughly corresponding to a sentence) from the user followed by some kind of response from the computer. Human/human communication is much more varied.”. Harris [32, p.597] also highlights the need for correct subjectively perceived context in identifying effective communication, stating that “communication is most accurate if the sender correctly anticipates the context in which the receiver attempts to understand the communication”.

Edlund et. al. [33, unformatted] considers how users of a computer-based speech interface may "perceive spoken dialogue systems in terms of other, more familiar things" and suggests that possible examples for such metaphors could include a human, a machine or an intermediate state. This suggestion aligns with the findings of Bradford [31, p.65] and also indicates that 'natural language' in an HMI context may not necessarily be the same natural langue used in a human-human interaction.

The degree to which using human-human interaction as a basis for human-machine command processing would degrade performance and thus user experience is not clear. The material reviewed did however show that recorded human-machine interaction is a considerably more representative data set for processing of natural language human-machine speech.

### 2.4.1 Perceived voice interface capability

The effect of perceived voice interface capability is described by Harris et. al. [32, p.597] who stated that “if the sender expects the receiver will have more or less information than is actually available, communication suffers”. If therefore the process of generating information on which command processing is based engenders a different perceived reader capability that would be the case when commanding a speech interface, this suggests that command phrases may be adversely altered. Differences due to this effect are most likely to manifest when comparing material based on the assumption of a human recipient with a good world view and pattern matching capability and commands given to a computer with significantly less capability for both factors. This is supported by Amalberti et. al. [9, p.4] who ask "can results, dialogue or user models obtained from the observation of human-human dialogues or from controlled experiments, be applied to human-computer communication?" and summarise a range of literature, identifying that speaking with machines generates simplified language due to:

> “- less dialogue control acts: less read-backs, less acknowledgments, less expressions aiming at closing or maintaining the communication;

> - less dialogue structuration: less linkage between the themes decrease in the role of request markers (markers are less numerous, less diverse, and meaningless); however, there is an increase in explicit communication planning;
>
> - attempts to "normativize" the language (i.e. production of a language closer to written language): less indirect requests, less uncompleted sentences, more well-formed requests;
>
> - attempts to simplify the expression: less contextual expressions (e.g. anaphora, ellipses. pronouns), repetitive use of the same syntactic expressions, reduction of the verbal group, smaller vocabulary.
>
> - attempts to simplify the communication: shorter conversations, decrease of the speech rate or volume, increase in the number of interventions, less chaining between the interventions (longer pauses, less interruptions, less overlaps), less digressions and comments, less expressions of politeness."

Baber [34, unformatted] supports these concerns, stating that "naturalness' results as much in the context in which speech is used as in the characteristics of speech itself" and continues to ask "if we use speech in a novel or unusual context, will speech remain natural, that is, is it natural to speak to computers?". Baber also notes that "subjects' speech to humans was verbose and contained several 'dialogue management expressions, such as 'er', 'ok', whereas subjects' speech to the computer was short, succinct and task specific; using simple imperative commands and used a restricted vocabulary".

### 2.4.2 Regional syntactic variation

A command processing method which relies on matching a new command to previous examples of commands intending the same action may be sensitive to the similarity of the command phrasing used by the people who gave example commands to that of a user issuing a new command. If therefore people from different geographic regions use different command phrasing for commands whilst intending similar actions, it may be necessary to build a corpus of samples from the regions from which each user originates in order to achieve acceptable speech interface performance. Spruit [14, p.8] assessed samples from 21 regions of a Dutch language area and performed a regression analysis of distance against syntactic variation. This showed a correlation coefficient of 0.55 and whilst statistical confidence was not shown, Spruit concludes that there is "geographic cohesion in syntactic variation", i.e. that syntactic variation correlates to some degree with geographic distance".

### 2.4.3 Corpus modality effects

One option to populate a corpus of command examples is to use one or more different modalities to collect the information. It is possible however that modality difference could result in differences in command phrasing which could adversely affect action identification.

Bigot et. al. [28, p.1051] supported this concern, noting in a natural language interface study including both a typed and spoken modality that "Spoken utterances were longer than typed statements" and therefore differed in phrasing. However, Baber et. al. [7, p.56] noted that "choice of words for performing even quite simple tasks with speech recognition is influenced by a number of factors" and identified one such factor as "The use of a speech recognition device leads to a reduction in the average length of utterances". One possible explanation for the difference is that the former study refers to processing of commands with different modality whereas the latter point refers to recognition of spoken commands rather than processing.

The effect of differing modality on language was considered by Amalberti et. al. [9, p.7] who performed an experiment considering human interaction with an electronic mail system and concluded that "typed interaction increased the tendency to normativization of the language". This difference in typed and spoken language suggests that basing natural language command identification on a corpus of a modality which is different to that used in operation may be less effective that using a common modality. However, it does not suggest the degree or significance of this potential effect.

Zoltan-Ford [18, p.538] notes that, while speakers model the length and content of their response on that provided by the computer, this effect was found in both speech and keyboard conditions. She also notes the need for processing care for phrase-length calculations as the number 21 could be considered as one or two words depending on coding. This is an important point where a typed modality corpus could be used to pattern-match a spoken modality command.

Zoltan-Ford [18, p.545] suggests that voice users allow prompting for details whereas typed users will take care to formulate a complete instruction. Whilst this does not fully align with the preference against dialogue expressed by participants of study 2, the difference may be explained by the dialogue (in the case described by Zoltan) referring to staged building of a complete command whereas dialogue (as described in study 2) refers to the selection of an action where a command was unambiguous but was not fully understood by the system due to limitations in the command processor.

In a data entry modality study, Luximan et. al. [35, p. 140] note that "traditional, keyboard and mouse input was better than voice input in terms of satisfaction and performance (task time)". They note that subjects who used a tonal language (in this case Cantonese) for voice input had the highest error rate, lowest satisfaction and longest task times. This may however be more attributed to voice recognition difficulties caused by the tonal nature of the language (where different tones indicate different meanings) rather than suggesting that typed and spoken modalities differ in choice of command phrase.

Goodkind and Rosenberg [36, p.88] noted that "A typist must undertake the cognitively demanding task of text production" and note that "The typing task is especially daunting for novice typists." This could result in a correlation between typing experience and commonality of typed and spoken modality command phrases and this is assessed in chapter 4.

### 2.4.4 Measurement methods

One consequence of the observation that context is a significant factor in gathering representative command samples is the decision against using a Wizard of Oz (WOz) method (where a concealed human operator provides responses which are believed by participants to be those of a computer). Whilst Dahlbäck & Jonsson [37, p.291] note the lack of a fully developed man-machine linguistics and suggest that one way of tackling this problem is to "simulate man-machine dialogue by letting users communicate with a background system through an interface which they have been told is a natural language interface, but which in reality is a person simulating such a device", Baber et. al. [7, p.48] summarise a range of literature by stating that "the (Wizard of Oz) method has been criticised as providing a different communicative context to real speech- based action with machines". The WOz method is further discussed in Chapter 4.

Ai et. al. [38, unformatted] studied the effect of differing measurement environments given the same study method. They found that dialogue obtained from study participants using a telephone speech-interface in a laboratory environment exhibited statistically significant differences to users interacting with the same telephone speech interface without being asked to do so from a controlled environment.

## 2.5 Habitability

Watt [12, p.338] introduced the term ‘habitability’ to describe the degree to which the language expressed by a user aligns with the language which a voice interface system can accept. Watt proposed (when referring to a question-answering device) that a system which can accept a single paraphrase for expressing a question might be termed ‘semantically adequate’ and goes on to show that such a system is not tolerably habitable for untrained operators due to the range of possible (semantically similar) phrases available to users. Furnas et al. [10, p.964] note this as a particular problem for new or intermittent users of computer systems, defining the “vocabulary problem" as consequent use of the wrong words for interface objects or actions. They performed empirical research which demonstrated considerable variation in (untrained) users’ choice of words and suggested an 80-90% failure rate for semantically adequate systems. Furnas et al. suggest that a single access word can only be expected to cover a small range of users’ choice of command phrasing and align with Gould [39, unformatted] in suggesting that designers underestimate this problem.

Baber [5, p.638] defines syntactic constraint as the number of different paraphrases which are allowed (by a system) for expressing a given meaning and describes a ‘fully habitable’ system to be one which can accept every possible paraphrase. Syntactic habitability is defined by Watt [12] as the ability of a system to accept paraphrases which will be spoken by users; a subset of all possible paraphrases. This thesis describes methods to use example command phrases to identify the actions intended by spoken commands and is therefore focussed on the degree to which such a system (comprising example command phrases and command processing method) approaches syntactic habitability. This is further considered in the introduction of chapter 4.

Ogden [40] notes that as a system is made less habitable, constraints are placed on what users can input into a voice interface. From the above definitions, it can be surmised that the degree to which syntactic constraints impact habitability relates to the likelihood of the removed phrases being selected by users.

Habitability can be improved by training users to work within system constraints, selecting paraphrases which better match user’s command choices or increasing the number of system-acceptable paraphrases. Increasing the number of paraphrases also increases the number of syntax branches which is defined by Schermant [41] as perplexity. Schermant suggests that increasing perplexity may reduce speech recognition accuracy (presumably because there are more similar sounding words which may be aliased). Command recognition which is based on pattern matching rather than a simple list of acceptable alternatives may be similarly affected by the increased number of potential phrases.

There is therefore a trade-off between highly habitable, low syntactic constraint systems which may present difficulties for speech recognition and potentially command identification systems and low habitability, high syntactic constraint systems which present challenges to users who are limited to specific system-acceptable phrases. Green and Wrede [42, unformatted] state that users employing 'out of vocabulary' words are a prominent problem and that this results from users being unaware of system capabilities. Hone and Baber [11, p.88] note that lower recognition accuracy (which may result from out of vocabulary utterances) resulted in increased transaction times due to the need for error correction.

Baber [5, p.638] notes that habitability can be improved (without reducing syntactic constraint) by either identifying a set of paraphrases which people will actually use and/or methods to inform users of the available phrases. Watt [12, p.339] defines an aspect of the habitability problem as 'determining criteria by which we can judge how likely a sentence is to be submitted'. Users may be informed of available phrases in advance (involving users memorising commands) or by viewing a screen based list (an eyes-off-road safety issue in an automotive environment). Baber [5, p.657] demonstrated however that use of a visual display to show all the currently valid speech inputs is not sufficient to ensure a habitable system as participants in a study sometimes deviated from the syntax displayed on screen.

In the case of highly habitable systems with low syntactic constraints, one method of identifying a set of paraphrases which people will actually use may be to apply a rational sub-grouping approach to optimise a system for each specific user. Zudilova et al [17, unformatted] consider user-specific optimisation via adaptive vocabularies. Pre-usage user-optimisation or in-use adaptation would increase habitability for a given number of paraphrases since the selected paraphrases would be those which a specific user is likely to select. Based on the relationship between a user's geographic origin and their syntactic dialect (noted by Spruit [14]), one method of rational sub-grouping considered in this thesis is the potential to obtain geographically tagged information and employ a selected subset of that information to identify commands given by a user of known geographical origin.

One aspect of habitability described by Baber as semantic constraint [5, p.640] relates to users and system designers having a common understanding of the application domain and by inference, the limits of functionality. Achieving such understanding may be particularly difficult in an automotive environment. For example, a system designer could include a function to turn on the vehicle headlights but may exclude a function to turn the headlights off as accidental use of this function could be a safety issue. Whilst headlight control would probably be excluded entirely to limit confusion in this case, the example shows the types of considerations incumbent on an automotive system designer.

## 2.6 Methods for identifying the action intended by a voice command

Speech interfaces can be broadly divided into those where allowable command phrases are to some degree constrained and those where users can express a command using any unambiguous phrase. This thesis is concerned with the latter option; natural language interfaces. In considering natural language processing, Magerman [43, p.1] notes that ambiguity is the basis for difficulty in natural language parsing and identifies that this is due to the effect of "context, the speaker, and many other factors".

### 2.6.1 Key word methods

Key words may be selected by a system designer or may be automatically extracted from available speech material as described by Hulth [44]. Key word systems attempt to identify the intent of a command by looking for specific words and discarding others. White et. al. [45, p.52] describe key word methods as both naïve and also due to their 'accessibility and economy', the most popular. Magerman [43, p.11] noted that "keyword analysis and pattern matching programs solved a small part of the natural language processing problem, but provided no mechanism to solve the remainder of it.". That discarding non-key words could reduce chance of command identification was noted by Bradford [31, p.62] who stated that "language itself provides many redundant clues as to what is said".

### 2.6.2 Pattern matching methods

Pattern matching methods can be more complex than simple key word processing. Sentence fragments of differing patterns may be compared in order to identify degree of match and/or semantic primitives may be matched rather than words. Automated methods of identifying patterns in text at a word or sentence level are described by Erkan & Radev [46] who used pattern-matching to summarise the most important sentences in a document and Mihalcea & Tarau [47] who used a linked-text ranking model to identify importance of words and sentences. One issue with methods to automatically isolate important words or sub-sentence fragments is in knowing whether the source material is relevant to the purpose to which it is to be put and this is considered in section 2.5.4.

Zoltan-Ford [18, p.527] states that "There are two ways to maximize the probability that a natural-language processor will understand each user input: (1) program the computer to understand the many ways people can structure their inputs; or (2) curtail the variability in people's inputs". The reference to curtailing variability in peoples' inputs could be interpreted in a relative sense by ensuring that people's inputs are similar to that which a processor expects. Spruit [14, p.8] notes that syntactic content is linked to geographic location and therefore basing a processor on information from the geographic region of a user may in effect meet this criterion.

Zoltan-Ford also states that “the developer of a natural-language processor must anticipate not only which words people will choose, but also how these words will be strung together". The reference to input structure supports the view of Bradford [31, p.62]; that there is information in command phrases which is not directly related to functional terms. An example of the relevance of this observation can be considered by comparing the commands 'headlights on' and 'turn the radio on'. In this case, the redundant term 'turn' would indicate that the subject is the radio. Should such patterns occur, they could enhance command identification where the subject of a command would otherwise not be recognised. The need for pattern-matching methods to accommodate a potentially wide variety of command phrases and to include language features such as concatenation is described by Kwasny & Sondheimer [48].

Zoltan-Ford [18, p.528] conclude that “There will always be one more way for a user to phrase a request that the designer did not anticipate” which further supports the view expressed by Gould [39, p.97] that data needs to be collected from actual users rather than contrived by system designers.

### 2.6.3 Large corpus statistical modelling methods

The scope of this thesis covers automotive command-driven voice interface systems and considers methods which make use of data which can be pragmatically obtained across a wide geographical region. Statistical natural-language processing instead involves using a machine-learning approach to process large training corpuses of text. Large corpus processing allows a system to model keywords and more complex sub-sentence fragments. The process involves passing large annotated text corpuses to an algorithm which both extracts key-words (sometimes based on word occurrence frequency) and also looks for other characteristics such as grammar, punctuation and lexical similarity between phrases. In addition to the need for large amounts of data to operate, Lui et. at. [49, unformatted] note that whilst these methods work well with pages or paragraphs, they are less effective on sentences.

Methods involving the construction of weighted grammatical structures from large corpuses of material as a way of statistically modelling natural speech have been and are subject to a great deal of research. Such methods are however out of the scope of this thesis which considers the use of simple pattern-matching methods supplemented with highly user-representative command samples which a crowd-sourcing approach has the potential to facilitate.

### 2.6.4 Data collection methods

Data gathering methods referred to in this thesis include vehicle simulator studies, street questionnaires, use of pre-existing material and crowdsourcing. The usefulness of these methods was considered from the perspective of obtaining command samples from many participants across many regions and countries.

Since simulator studies require simulators and the experts required to run them and given that street surveys require people to undertake this task, resource required to gather samples using these methods must be related to the number of samples required. The value of using pre-existing material for a command processor corpus depends on whether the material used employs a common syntax with that used by people giving commands to an automotive voice interface. Amalberti et. al. [9, p.7], Baber [34, unformatted] and Amalberti et. al. [9, p.4] all found that speech to computers does not match human-human speech and this supports the need for command samples which are context relevant.

Crowdsourcing is a possible way to alleviate context-relevance and cost/resource challenges. Behrend et. al. [50, p.801] defined crowdsourcing as "the paid recruitment of an online, independent global workforce for the objective of working on a specifically defined task or set of tasks". In the context of this thesis, data samples will be relevant whether provided in response to being paid, for interest or because the data is obtained via a means designed to entertain (for example a computer game). However, that crowdsourced data is both independent and global support the use of this method for obtaining a command processor corpus suitable for use across a broad geographic area.

Behrend et. al. [50, p.808] also considered whether crowdsourcing produced data of a quality comparable to a selected (university) sample of study participants. They concluded that "data were of equal or perhaps better quality in the crowdsourcing sample".

## 2.7 Speech interface design

In the case where a voice control system receives a command which is unambiguous and where the action is within the remit of the system, no response may be needed beyond enacting the command. In this case, the human-machine interface could be limited to a simple spoken command −> enacting-of-command framework. It could therefore be envisaged that where a speech system needs to take a different action to enacting a command, it is in effect handling an error state. Factors affecting the quality of human-machine interaction therefore include:
1. The percentage of commands the system can identify and enact.
2. The type of response used where the system is unsure of the intended command.

### 2.7.1 User commands

One explanation for why fixed command systems may require command phrases which are different from that which users may prefer is noted by Hua and Ng [51, p.31] who assess an existing commercial system and state that "The current commands are also designed to sound very different from each other. This was done to increase the recognition success rate. However, for commands that are linked functionally, this sounds unnatural to the user and also requires the user to remember extra commands".

Bradford [31, p.61] argues that “in the domain of speech recognition … the recognition of user action is inherently error prone”. This proposes that it is not therefore possible to avoid the need for error handling. Bohus and Rudnicky [52, unformatted] quantify this for one circumstance, saying that under the circumstances of "spontaneous speech, large vocabularies and user populations, and large variability in input line quality" that "average word-error rates of 20-30% (and up to 50% for non-native speakers) are quite common". Kamm [53, p.10031] notes that "A carefully crafted user interface can overcome many of the limitations of current technology to produce a successful outcome from the user's point of view, even when the technology works imperfectly". This point suggests that efforts to improve voice interface acceptance should be broader than simply looking to improving the percentage of commands which can be correctly identified; the user-interface for handling inevitable cases where a command is not correctly recognised also requires consideration.

Kamm [53, p.10031] also proposes a conversational interface stating that "With further technological improvements, the primary role of the user interface will gradually shift from a focus on adapting the user's input to fit the limitations of the technology to facilitating interactive dialogue between human and machine by recognizing and providing appropriate conversational cues". Boyce and Gorin [54, unformatted] support this point, saying that “We are interested in building machines which can understand and act upon fluently spoken language. This is often not a single interaction, but rather involves a dialog to negotiate the proper outcome”. These interactive interface considerations differ from the command-response focus of this thesis and may (in the case of Kamm) refer to contexts where a user may not know what action they require and need to discuss options or (in the case of Boyce and Gorin) relate to interaction with future artificial intelligence systems. Participants in study 1 (chapter 3) were not generally in favour of dialogue-based interaction for automotive voice command systems.

Where a command is not correctly identified on the first try, there may be a potential for user stress and consequent reversion to natural language. As noted by Baber et. al. [55, p.41] "Speech errors, which involve violation of vocabulary constraints in using speech recognition systems, could arise because users revert to more familiar words and phrases. This problem will also be apparent to anyone who watches sports programmes: as the action becomes more exciting, commentators seem to revert to more and more clichés". Bradford [31, p.64] notes that “With respect to the use of speech-based human/computer interfaces, the most important emotion that influences speech is apt to be frustration”. Chen and Saav [56, unformatted] show that giving commands under time stress reduces voice recognition accuracy. Tension, stress or frustration may result in voice modulation, rate of speech, speech clarity and/or changes in command phrasing which further reduce the probability that the voice system will identify a command. If an error handling interface was able to reduce negative emotional

response to commands which have not been recognised, it may therefore improve likelihood that further attempts will be successful.

To address both improvement of command recognition and also error handling to best effect on user satisfaction, knowledge of users in general will be required. In some cases knowledge of the specific user issuing a command may also be needed. However, Gould [39, p.97] observed that interface designers show "reluctance to define the users" and "a reluctance to take the definition seriously". He noted that simply considering the perspective of users was not adequate, saying "Even if a description of the intended users were as complete as it possibly could be, it could not be an adequate basis for design. It would not substitute for direct interaction and testing". This suggests that a representative sample of users would need to be directly consulted to accurately establish factors such as command phrases and thresholds for different types of response.

Gould [39, p.98] also states that it is not reasonable to "expect users to invent radically new ideas. Most users are not trained to think through a design but they can comment on your new ideas if you show the ideas in an appropriate form". Where users of existing systems were asked about their interest in natural language processing methods in study 1 (chapter 3), this was therefore moved into a context with which participants would be familiar, asking instead if users would define their own commands if they had the opportunity.

### 2.7.2 Command-recognition error management methods

Bohus [57, p.154] states that "experiments in two different dialog domains revealed that the impact of misunderstandings and non-understanding on performance is non-linear and the relative costs of these errors are different across domains". This suggests that it will be necessary to determine thresholds for system responses which are specific to an automotive context to optimally mitigate reduction in user-satisfaction under circumstances where a command processing system is unable to identify an intended action.

System responses are considered by Boyce and Gorin [58, unformatted] who describe four types as confirmation, disambiguation, re-prompt and bailout. They go on to describe confirmation as "the system repeats what it thinks was said, giving the user a chance to confirm or deny the system's interpretation". Whilst this is the only option for voice recognition alone, a natural language command interface could instead relate to the user the action which it identified as the most likely to be that intended given the command it received. This would then prevent the possibility that a user might confirm that the spoken words were correctly recognised, only for the command processing system to incorrectly ascribe an unintended action to these words.

Ambiguity is explained by Boyce and Gorin as "more than one interpretation of the user's speech" who then describes two options to handle this, either asking which of two options is intended or asking if one option is intended thereby indicating the other if the response is negative. One additional possibility is to consider the command given against the two actions between which the system cannot decide and identify what words would, if added to the command, most effectively resolve the ambiguity. These words could then be presented to the user to select from.

Re-prompts are described by Boyce and Gorin as "ask for the user to repeat his or her request". This action could be expanded into two possible versions depending on where in the voice command system the confidence was low. If the voice recognition system had poor matches of the sound patterns to the speech model, asking to repeat the command would be ideal. In the case where the command was effectively recognised but the command processing system was unable to identify the intended action, a request to rephrase the command would be more effective.

The 'bailout' option described by Boyce and Gorin proposes referring a user to a human operator in cases where a command is repeatedly not recognised. In an automotive context this is not possible, instead a user might be asked to use a conventional control (assuming a conventional control was available.).

Bohus and Rudnicky [52, unformatted] describe ten options for error-handling, including "Notify (system merely notifies the user that a non-understanding has occurred)" and "Yield (system remains silent, and thus implicitly notifies the user that a non-understanding has occurred)". In a random response study, they found that asking for confirmation of the most probable action was most likely to result in the statement immediately following being understood by the system. A verbose description of the context and phrasing which would enact specific actions was the next most effective. Whilst probability of subsequent success was one measure for effectiveness of an error recovery method, degree of user satisfaction with each interaction was not clear.

## 2.8 Literature review discussion

The literature reviewed in this chapter proposes compelling reasons for the development of voice interface methods. Whilst factors such as dashboard clutter could affect time to locate conventional controls and could degrade aesthetics, safety issues with conventional interfaces were most significantly highlighted in studies, particularly those relating to the duration a driver needs to look away from the road. Cognitive load resulting from the use of voice interfaces and the consequent effect on ability to prosecute other tasks (particularly the primary driving task) is not however so clear. Graham and Carter [59, p.31] note that during a study, participants showed reduction in sensitivity in performing a task while using a voice command system when compared to using a manual control. A study by Greenberg et. al. [23, unformatted] notes that gap to leading vehicle increased when using a hands free voice interface although there was not an increase in road position error or lane violations (noted as "a lane exceedance large enough to move one of the vehicle's tires completely outside the left or right lane markings"). Another study found that speech command recognition accuracy affected steering wheel angle variance very significantly but not lane position or velocity. One possibility is that drivers were aware of the reduction in finesse of control due to cognitive load and adjusted their available reaction time to maintain quality of driving. Graham and Carter [3, p.160] used a tracking test to show that lane edge collisions occurred more commonly with conventional interfaces than with speech interfaces.

Several methods for identifying a action intended by a voice command were noted in the reviewed literature. Simply matching a fixed phrase to an action is a low habitability method which is in-use in current commercial systems. However, this requires users to memorise precise commands, something which Baber et. al. [55, p.46] shows may become less reliable when a user is in a stressful situation. It is also vulnerable to words being incorrectly recognised as any incorrect word will result in the command not precisely matching the pre-set phrase. A more habitable system which allows users to employ any command phrase they wish presents a natural language processing problem for which several methods were also described. Key word matching was noted as of limited capability and is therefore unlikely to form the basis for automotive voice interface methods development.

Pattern matching methods require a pattern to match against and it can be seen from the literature reviewed that this corpus of information can originate from and be used in a number of ways. Machine-learning methods may be used to develop a grammar of word-sequence probabilities by observing patterns in human speech. Baber [34, unformatted] notes that spoken statements in the context of voice control tend to be "short, succinct and task specific". It is possible therefore that material used for machine learning may, if from a different context, have different syntax to that generated in an automotive voice interface context. Obtaining

material in an ideal context presents the problem that, as a capable automotive voice interface has yet to be achieved, the context of interacting with such a system cannot be directly realised.

Whilst such a system might be simulated using a Wizard of Oz method (where a concealed human takes the place of a computer system), limitations with this method are noted by McTear [60, p.147] who point out difficulty for a human to behave precisely in the manner that a computer might and "to anticipate the sorts of recognition and understanding problems that might occur in the real system". One alternative to the three methods noted in the reviewed literature (fixed phrase, key words, machine learning) would be to compare components of a command to a corpus of commands given by people who use similar command phrasing.

Since dialect varies with region (as noted by Spruit, [14, p. 8]) as does language, pattern matching to example commands which use similar command phrasing would require a method of obtaining a sufficient number of such example commands from potentially a global geographic area within a reasonable level of resource. Using people to gather such information would scale resource requirements with area and could quickly become prohibitively costly. One option to perform the task in a way which scales without additional cost would be to collect samples via the internet in a typed form. Research shows that changing modality will affect language but not to what degree.

An ideal voice interface could simply do as instructed with the enacting of the command the only confirmation needed. One use of a voice command system interface is to manage circumstances where the system does not have sufficient understanding or confidence in understanding of a command to enact it. Several academic papers consider the types of errors which can occur in interpreting a spoken command into an action. The different types of system response to a user for each type of error are also considered. However, research did not identify methods for devising optimal confidence thresholds for selecting an optimal response.

## 2.9 Conclusion

Literature reviewed in chapter 2 shows that much research work related to aspects of voice interfaces has been completed and is ongoing. Opportunities have been noted for research into the development of methods within the scope of the question stated in section 1.4. Figure 2.1 shows a high-level operation sequence for a voice interface and highlights the sections for which additional research opportunities were noted.

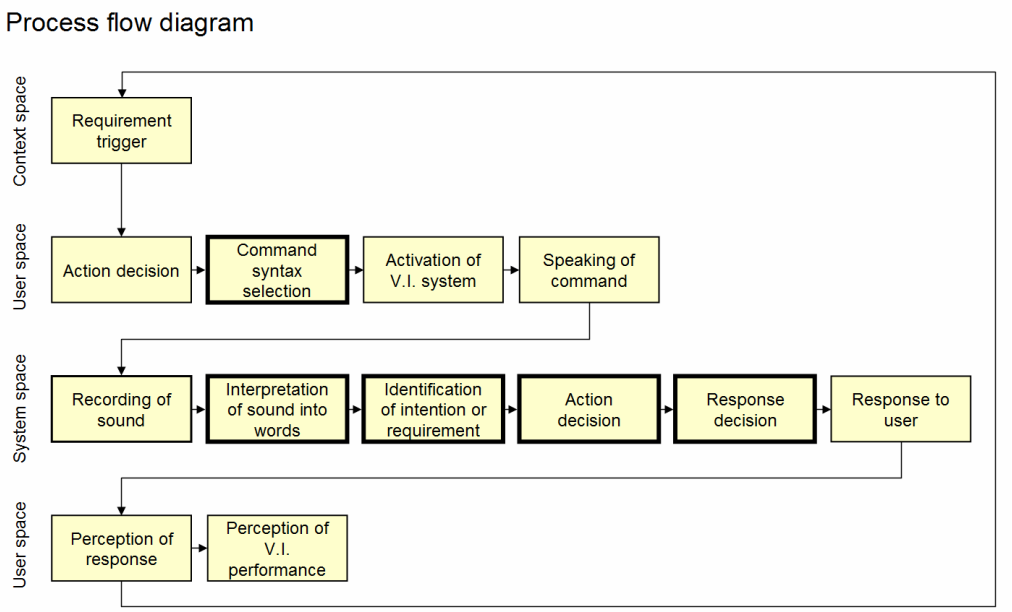



Figure 2.1. Voice interface process flow-diagram highlighting key areas of research.

Study 1 (chapter 3) comprises a semi-structured survey which considers user experiences with current voice interfaces.

# *Chapter 3: User perception of current automotive voice-interfaces (study 1)*

## 3.1 Introduction

### Aim of Study 1

Study 1 considered the current state of automotive voice-command interfaces to understand:

- How satisfied study participants were with current automotive voice-interfaces
- Where satisfaction was low, the factors which contributed to this.

If habitability were shown to adversely affect user-satisfaction, further research to develop methods to obtain data which allows users to employ their choice of command phrasing would be supported (questions 1-2, section 1.4).

### Overview

Voice interfaces are used commercially in many areas; examples include automated telling machines, telephone menus and automotive systems. The capability of such systems has been considered in researched literature such as Lo and Green [61] who described a range of user-verbatim noting issues with automotive voice interfaces and Oulasvirta et. al. [62] who identified a correlation between frequencies of errors and usability ratings. Commercially available voice interfaces are developing over time and obtaining a clear understanding of current user opinions required that existing research be extended by understanding users' experience of more recent systems. In study 1, a semi-structured interview was performed in order to evaluate participants' experience of commercially available voice interfaces. During study 1, it was possible to include questions relating to the value which users may place on an increase in system habitability and specifically a reduction in constraints on command phrasing.

Questions addressed both participants' overall satisfaction with the voice interfaces they had experienced and in cases where they were aware of aspects of voice interfaces which affected this satisfaction, the degree to which each contributed. Questions were based on factors identified during the literature review described in chapter 2, particularly on Hone and Graham [63, p.2] who noted six categories of interest comprising accuracy of interface response, whether users like an interface, the mental workload required to operate a system, how annoying a system is to use, whether a system met expected capabilities and how quickly commands could be enacted using the interface.

## 3.2 Participants

The subjective opinions of people who had prior experience of using one or more current commercial voice interfaces were gauged. As recall of usage experience could become degraded after an extended duration of non-use and limited use could result in views which are not fully formed, participants were limited to those who, at the time of the study both owned a

vehicle fitted with a voice-interface system and who had owned a vehicle with a voice-interface (although not necessarily the same vehicle) over the year preceding the study.

Twenty participants who met the above criteria were selected for the study, covering both genders and from a 20 to 60 year age range. The vehicles owned by participants came from a range of manufacturers including BWM, Jaguar, Ford and Land Rover.

## 3.3 Method

A semi-structured interview method was selected to enable a common set of questions to be put to participants whilst allowing them to expand on answers with additional ideas or thoughts thereby developing a more information-rich data-set.

High level topics for the interview questions (Table 3.1) were identified from the contexts and sequence of operation in which users may interact with a voice interface. These include interaction with a training system, submitting a command to a system, receiving a system response and circumstances where a system does not respond as desired. Study factors (Table 3.1) which related to user experiences within each context were identified and a set of specific questions was developed to gauge user-responses against these factors (Table 3.2). Participants were also asked to rate on a scale of 1-100, aspects of the voice interface systems of which they had experience and were also asked to suggest potential improvements. At the end of the interview, participants were shown an image of a television remote control and were asked to state the verbal command they might use to enact commands equivalent to five indicated buttons on the remote control.

One potential question related to whether participants would prefer a natural language voice interface to a fixed-command interface. However, Gould [39, p.97] observes that whilst people are not generally trained to think through a new design concept and cannot be expected to invent new ideas, they may be able to comment if presented with a new design concept in ‘concrete and relevant’ way. The alternative question ‘If you had the facility to teach the system some commands you want to use, would it be worth the time?’ (Q11) was therefore substituted to allow users to indicate if they had preference for natural language within a context relevant to their experience.

A potential biasing effect included the use of leading questions in the manner described by Schuman & Presser [64, p.152] who noted in relation to a survey on allowing speeches against democracy that approximately 20% more people preferred to ‘not allow’ when compared to those who chose to ‘forbid’. Questions were constructed to minimise use of biasing or leading words in questions, for example replacing ‘did you like’ with what was your opinion of’

A second potential biasing effect was the possibility that earlier questions might condition responses to later questions as noted by Zaller and Feldman [65, p.600] and Bertrand & Mullainathan [66, p.67]. The sequence of questions was arranged to minimise this potential by initially asking open, non-leading questions and later asking questions which were more specific and therefore might affect responses to the more open questions.

Table 3.1. Identification of study factors from high-level topics (study 1).

| High-level topic | Factor | Related study factors |
|---|---|---|
| General information | 1 | Participant factors (age range, gender etc.) |
| Baseline voice-interface rating | 2 | Overall participant assessment of voice-interface system(s) experienced |
| User training/experience effect | 3 | The value of training methods |
| | 4 | Potential for participants to improve perceived performance. |
| Command phrasing factors | 5 | Acceptability of pre-defined command phrases |
| | 6 | Participant memory of command phrases |
| | 7 | Number and type of commands participants can recall |
| | 8 | Similarity of a set of prompted test commands |
| Inability of system to enact a command | 9 | Perceived ratio of errors causes, incorrect command phrases vs. poor voice recognition |
| Effect of interface design | 10 | Effect of error management methods. |
| | 11 | Acceptability of dialogues to manage errors |
| | 12 | Whether a dialogue or repeating/rephrasing is preferred to manage errors |
| User response to errors | 13 | To what degree does repeated error cause a negative emotional reaction |
| | 14 | System factors which cause a negative emotional reaction |
| | 15 | Might improvements result in restarting lapsed voice-interface usage |
| Perceived opportunities | 16 | Additional voice control capability desired |
| User specificity of interface | 17 | Commonality of desired interface settings |
| | 18 | Acceptability of user-adjustable interface settings. |

Table 3.2. Semi-structured interview questions (study 1).

| Factor | Specific question(s) | Question |
|---|---|---|
| 1 | Name | 1 |
| 1 | Gender | 2 |
| 1 | Age range 17-25 26-35 | 3 |
| 2 | Amount of time using current system to-date | 4 |
| 2 | Have you ever had a previous voice system in or out of a vehicle | 5 |
| 2 | Generally, how worthwhile are the voice interfaces you have used. | 6 |
| 3 | How did you find the voice system training? | 7 |
| 4 | If you have become better at using the system over time, what have you changed about your usage to do this? | 8 |
| 5 | Did you find the commands the system expects to be fairly intuitive from the outset? | 9 |
| 5 | If you tried to use a new command, do you think the system would understand you? | 10 |
| 5 | If you had the facility to teach the system some commands you want to use, would it be worth the time? | 11 |
| 5 | What if by doing so you could significantly increase the likelihood the system will understand you even when obscured by noise? | 12 |
| 5 | Do you sometimes use different commands for the same thing? | 13 |
| 5 | Why? | 14 |
| 6 | How often does the system fail to understand you even though you know you have said a command or word it knows? | 15 |
| 6 | What do you think causes this? | 16 |
| 9 | Do you usually know when giving a command whether it will work or not? | 17 |
| 10 | How would you describe the system taking a wrong action compared to it asking for you to try again? | 18 |
| 10 | Which of the two happens most often? | 19 |
| 11 | When it does the wrong thing, how easy is it to undo and then do what you originally wanted? | 20 |
| 11 | What do you think about how the system alerts you that it doesn't understand? | 21 |

| | | |
|---|---|---|
| 11 | What would be the most expedient and discrete way for the system to alert you to the fact that it didn't understand? | 22 |
| 14 | Compared to how good it might be, how would you describe the system in terms of being sprightly or sluggish to use? | 23 |
| 14 | Compared to how good it might be, how would you describe the system in terms of its being too terse or too verbose? | 24 |
| 14 | How do you think you would find using the system in company? | 25 |
| 14 | (If relevant) What do you think might make it more comfortable to use in company? | 26 |
| 7 / 8 | Could you please list for me the commands you can recall | 27 |
| 12 | If the computer has to present you with options to choose from, is this still preferable to using buttons? | 29 |
| 13 | Have you experienced a time where the system repeatedly failed to understand you? | 30 |
| 13 | (If relevant) Were you particularly bothered? | 31 |
| 12 | Would you rather be asked to repeat a command or be prompted to go through a command step by step if the system doesn't understand you? | 32 |
| Voice interface ratings (0-100) | Overall satisfaction | 34 |
| | Speed compared to manual systems | 35 |
| | Speed compared to instructing a human being | 36 |
| | Effect on concentration compared to conventional controls | 37 |
| | Intuitiveness of the system | 38 |
| | Performance of the system compared to your expectations | 39 |
| | The training system | 40 |
| 16 | If your system had the ability to interact with the internet, can you think what you might use it for? | 41 |
| 16 | If your system could understand what you are asking and answer questions intelligently, what might you use it for? | 42 |
| 16 | What would you like to be able to do that you currently cannot (with VI) | 43 |
| 16 | Would you like to be able to text people by using your voice for example? | 44 |
| 17 | How confident do you think the system should be that it understands before it goes ahead and takes the action? | 45 |
| 17 | What improvements could be made in the way the system responds to you? | 46 |
| 17 | Imagine you are in heavy traffic and you ask the system to turn off the radio. It only has (level of confidence which generated a non do-it response above). What would be the ideal way it should deal with it? | 47 |
| 18 | If you could add user configurable options to your current system, what might you find worth changing? | 48 |
| 18 | How about changing system response to a colour flash or sound to indicate that the command wasn't understood? | 49 |
| 18 | What if you could change between a range of voices and response types? | 50 |
| 15 | If you booked your car into a dealer and the dealer informed you that your voice recognition had been updated to a significantly enhanced system, would you be interested to see how much better it became? | 51 |
| 15 | What might your expectations be? | 52 |

## 3.4 Interview process

Participants were interviewed during the daytime and on a one-to-one basis. Interviews took on average approximately 30 minutes to complete. Each interview commenced with a brief outline of the intent of the study. A consent form was completed and participants made aware of their right to withdraw at any point. It was also made clear that before publication, any reference to specific participant identity would be deleted. No participant elected to withdraw at any point.

The interview was informal and a minimal conversational interaction was maintained whilst ensuring that no leading, conditioning or prompting (beyond the designed questions) took

place. On a number of occasions, it was necessary to reframe a question to accommodate an individual participant's experience. For example one participant responded to the question "Did you find the commands the system expected to be fairly intuitive from the outset?" by stating that they had looked-up the commands they intended to use before using the interface and therefore had no expectation other than that it would work as written. The prompt was elaborated to "Did you find the commands stated in the manual to be fairly intuitive?"

In some cases, an earlier response made a later prompt inapplicable. For example, if a subject indicated that they had not experienced an improvement in their usage of the system over time, it was redundant to ask about causes for improvement. In such cases, the question was omitted. Where participants stated that a question was not relevant to their experience, their response was omitted from the result.

## 3.5 Results

Descriptive statements given by users for general worth of voice interfaces varied from 'Very' (useful) and 'Very good' to "Not" (useful), 'Didn't live up to expectations.' and 'Pretty disappointing'. Responses to other questions were reviewed for possible reasons for the differing statements and is was noted that participants who did not consider voice interfaces of worth more commonly stated that the pre-set command phrase was not intuitive. Where participants found voice interfaces to be of worth, examples of responses to the intuitiveness of pre-set command phrases include 'Yes with a caveat, some systems expect dial number, some call' and 'Yes - only basic command set'. In cases where participants did not consider voice interfaces to be of worth, examples of responses to whether the pre-set command phrases was intuitive included 'No', 'Not intuitive commands' and 'I had to look them up. Tried, didn't work, had to look up.' One anomaly with this trend was a participant who stated that voice interfaces were 'not' (useful) but in response to the intuitiveness of commands, stated that 'Some of them are. Yes'. Further review of responses revealed that this participant stated that they had a 95% failure of the system to understand commands even if worded in a way which should be understood. In response to a question on possible reasons for system failure to understand, the participant stated 'My elocution is poor, even when I try it still doesn't understand' which may account for the outlier in terms of correlation between perceived voice-interface worth and intuitiveness of pre-set command phrases.

In response to a question on whether a guess at a command might work, all but one participant responded negatively with responses such as 'No', 'Wouldn't attempt' and 'Zero'. The differing participant felt some degree of confidence, stating 'Only if I got the right words. Reasonable chance'. Combining the two phrases suggests that the participant uniquely felt able to guess the pre-set command phrases, however the same participant responded to a question on

intuitiveness of pre-set command phrases stating 'you need to know them.' which was somewhat contradictory.

Responding to a question on voice interface response-speed; all but two participants stated that they felt the interface was slow. For both participants who stated that the interface speed was acceptable, the proportion commands which were correctly understood was at the high end of the set of responses and the system was therefore more syntactically habitable for those users. One possibility is that the perception of interface sluggishness was affected by the need to manage incorrectly recognised commands and therefore perceived response time related to the overall time needed to take an action via the voice interface rather than any individual speak-respond cycle.

A context was presented to participants of a vehicle in heavy traffic and a voice-command to reduce radio volume where a voice interface had too low a confidence in correctly identifying the user's intention to enact the command. One possibility was that participants might perceive that, as loud music can be distracting and given the context of heavy traffic, it might have been better (possibly safer) for the interface to first reduce radio volume and to then confirm the correct action (returning to the previous setting if not confirmed). This would suggest that response thresholds might need to be context dependent. In the study, only one participant suggested this type of response, the other participants proposed a normal level of issue management. It is unclear whether this was because participants perceived no value in such context-based responses or because participants may not perceive the value of this option unless in a driving stress context.

One option to assign participant responses to categories thereby allowing comparisons to be made was a content analysis method such as that described by Elo and Kyngas [67]. Another possibility was to extract and compare several measures such as direction, degree and emphasis. However, a review of participant verbatim showed that a simple positive/negative five-level scale could be applied to a majority of responses. Where questions elicited responses which could be evaluated on a range from a positive to negative view of the specific voice interface factor, they were coded using the relationship shown in Table 3.3. Figure 3.1 shows the result categorised in this manner.

Table 3.3. Coding method for semi-structured interview responses (study 1).

| Content of participant response | Example | Coded value |
|---|---|---|
| Unqualified positive | Excellent, worthwhile | 5 |
| Positive with limiting qualifier | Generally good, sometimes worthwhile | 4 |
| Indeterminate or not referring to intended measure | Interesting to try, feels different to conventional interface | 3 |
| Negative with limiting qualifier | Often poor response, not very useful | 2 |
| Unqualified negative | Useless, ineffective | 1 |

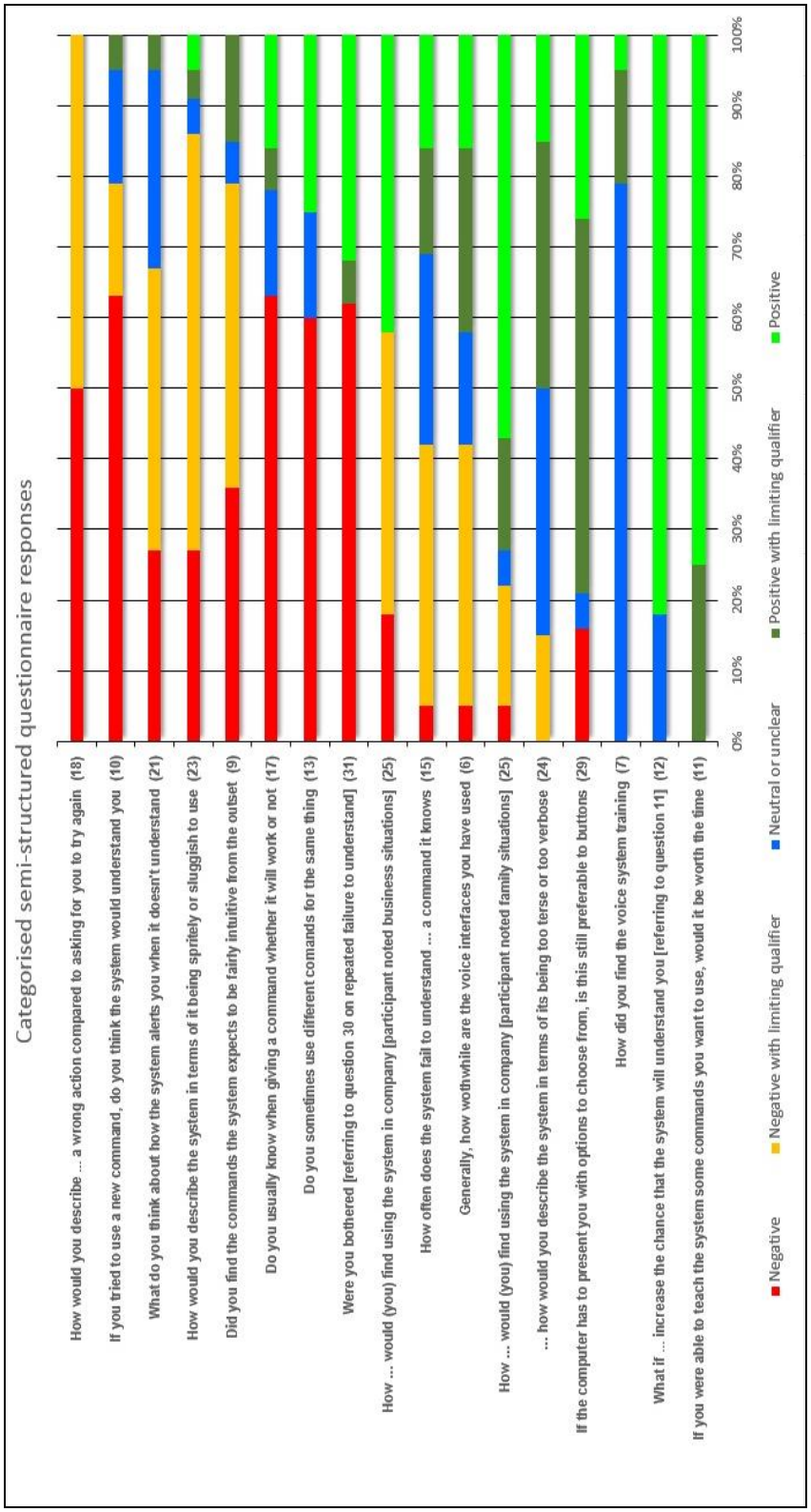


Figure 3.1. Categorised & normalised qualitative interview responses

Eight questions which prompted qualitative responses could not be categorised according to Table 3.3 and are described with individual categorisation of results. Question numbers relate to the question-set detailed in Table 3.2.

Question 8: If you have become better at using the system over time, what have you changed about your usage to achieve this?

| | |
|---|---|
| Syntax change | 7 |
| Pronunciation change | 7 |

Question 22: What would be the most expedient and discrete way for the system to alert you to the fact that it did not understand?

| | |
|---|---|
| Verbal | 14 |
| Non-verbal | 0 |

Question 26: What do you think might make the system more comfortable to use in company?

| | |
|---|---|
| Ability to use a natural tone of voice | 10 |
| Don't play-back users voice | 1 |
| Work more often | 1 |

Question 32: Would you rather be asked to repeat a command or be prompted to go through a command step by step if the system doesn't understand you?

| | |
|---|---|
| Definitely repeat | 7 |
| Prefer repeat | 5 |
| Neutral / unclear | 3 |
| Prefer dialogue | 1 |
| Dialogue | 3 |

Question 51: After a dealer upgrade, would you try your voice-interface again?

| | |
|---|---|
| Yes | 13 |
| No | 0 |

Question 52: What might your expectations be?

| | |
|---|---|
| Enacts command correctly more often | 5 |
| Controls additional features | 5 |
| Switch to natural language | 1 |
| Quicker operation | 1 |

Question 46: What improvements could be made in the way the system responds to you?

| | |
|---|---|
| Say my name | 1 |
| Be more polite | 1 |
| Match response word-rate to command word rate | 1 |

Question 45: How confident should the system be that it understands before it takes the action?

| | |
|---|---|
| Very / Pretty certain / Absolutely / 95-100% | 12 |
| 80% | 1 |

Questions 34 - 40 required participants to answer with a percentage value. Where questions compared performance to a benchmark, participants were asked to consider 50% to be equal performance and over 50% scores to indicate a better rating. The results did not show a good match to a range of parametric models (for example, a normal or Weibull distribution) so the data is shown as a scatter plot in Figure 3.2 with question prompts summarised. Individual (red) markers show participants' percentage ratings and the connected markers (blue) show mean values for each question.

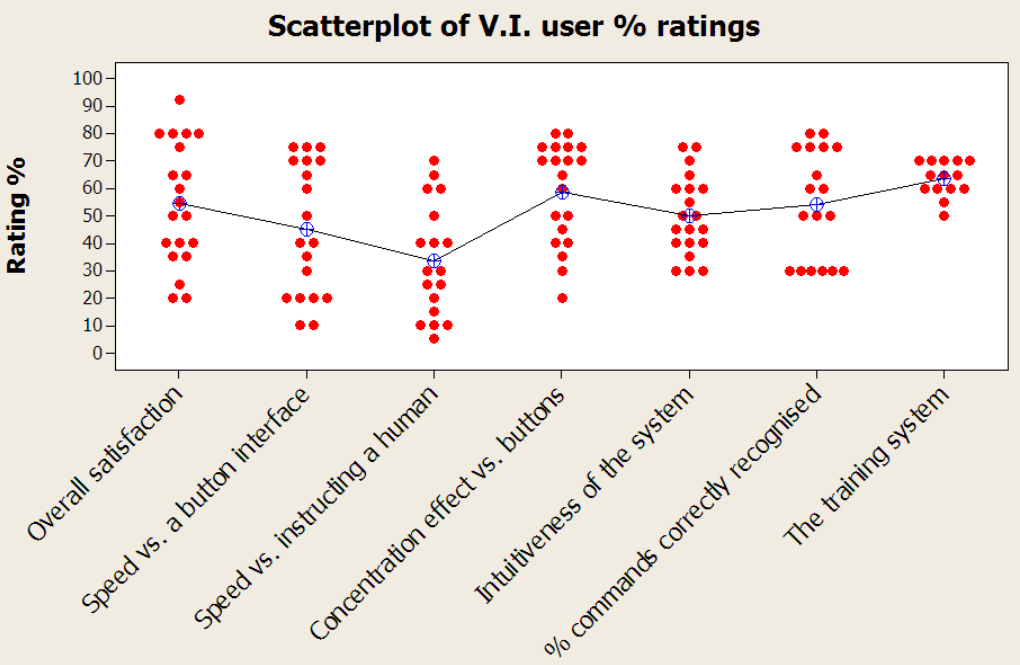



Figure 3.2. Scatter plot of percentage ratings for quantitative interview questions (study 1).

A multiple regression analysis was used to assess correlation of the factors shown in Figure 3.2 with overall satisfaction. The percentage of commands correctly recognised was the only significant factor [$F(1,16) = 47.98$, $p < .001$] with an $R^2$ of .752. Figure 3.3 shows this regression with confidence and prediction bands.

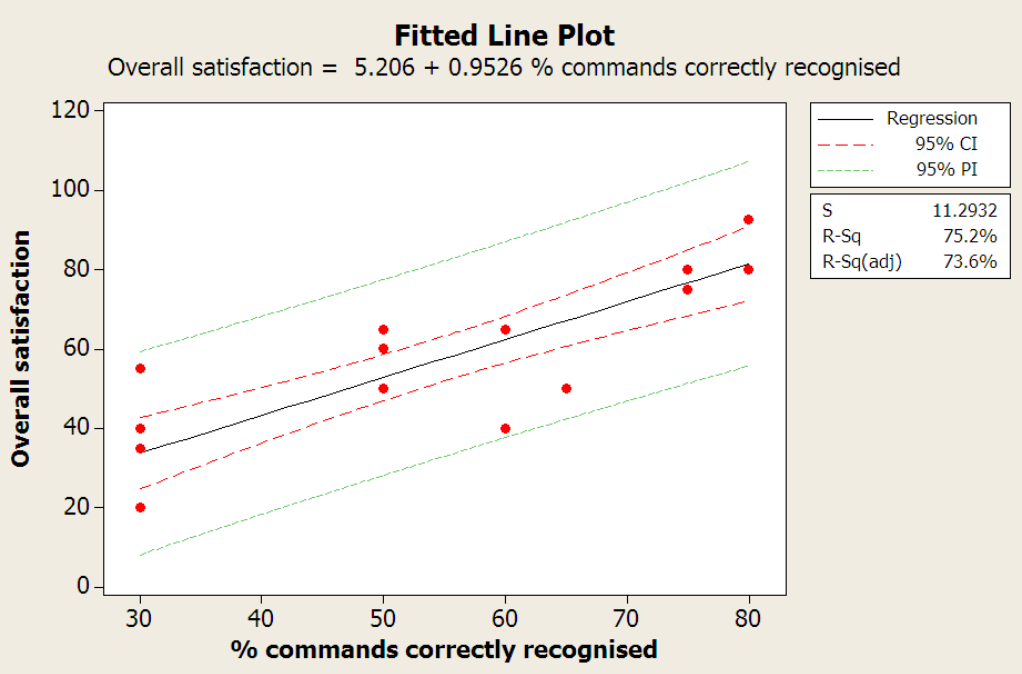



Figure 3.3. Correlation between percentage of commands recognised & overall satisfaction (study 1).

An image of a television remote control was shown to participants using the image shown in Figure 3.4. Participants were informed how to proceed using the words 'Assume that your TV is voice controlled and is currently on. Please say the command you would use in place of pressing the button shown.' Table 3.4 shows commands given by participants. The quantity values show the number of participants which used a particular command.

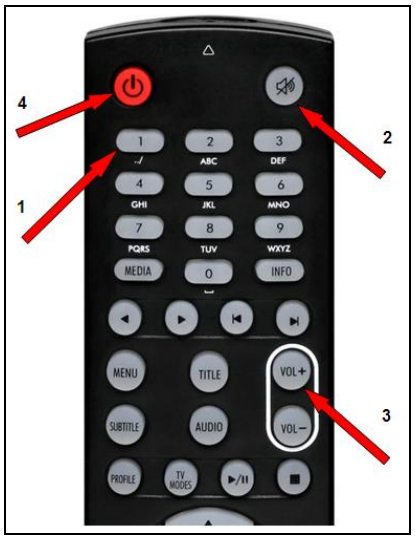



Figure 3.4. Prompting image for participants voice-commands

Table 3.4. Stated command phrase with participant counts (study 1).

| Actions | Change station to channel one. | Count | Mute the sound. | Count | Record a programme. | Count | Increase brightness. | Count |
|---|---|---|---|---|---|---|---|---|
| Responses | Channel one | 6 | Mute | 9 | Increase volume | 5 | Power off | 4 |
| | One | 3 | Volume mute | 2 | Volume up | 6 | Off | 6 |
| | Station 1 | 2 | Mute sound | 2 | Turn volume up | 2 | Power | 2 |
| | Channel select BBC1 select | 2 | Sound off | 3 | Volume increase | 3 | Turn off | 2 |
| | Select Channel one | 2 | No response | 2 | Increase volume three bars | 1 | Television switch off | 2 |
| | Change to channel one | 3 | | | No response | 1 | System off | 2 |

## 3.6 Discussion

An overall participant view of voice interfaces was obtained via Question 6 which asked participants to state their view of how worthwhile they found voice interfaces. Participant views were broadly distributed from very negative to very positive with a median value close to neutral, the distribution of percentage scores for 'overall satisfaction' (question 34) also had a median score of close to 50%. Both results indicate an opportunity to improve user satisfaction and perceived value of voice interfaces and thereby increase the degree to which the benefits of voice interfaces described in section 1.1 can be realised.

Results for other questions and logical groups of questions suggest three factors that may affect user satisfaction with voice interfaces.

Q9, 10, 11: For all three questions relating to pre-defined command phrases, participant responses were biased towards considering pre-defined phrases to be different to that which they would prefer.

Q15: Overall satisfaction increased with percentage of commands correctly recognised (Figure 3.3).

Q18: All participants gave a negative response to a voice-interface system taking an unintended action compared to asking for a command to be repeated or rephrased.

Possible reasons for users employing semantically valid command phrases which a system does not expect and cannot interpret include that they do not know the expected command phrase, they cannot recall the expected command phrase and the potential for a secondary effect such as driving stress causing a user to revert to a more natural command phrase. Baber and Noyes [8, p.144] note the possibility that people may, despite training, revert to previously learned patterns of behaviour when placed under stress. Figure 3.5 shows how low habitability in the form of fixed command phrases which are not natural for users may adversely affect the perceived value of voice interfaces and highlights how failing to advise the

user of the cause of action-identification difficulties may result in a repeated error loop with the potential to further reduce user satisfaction.

Figure 3.3 shows a statistically significant correlation which supports Casali et. al. [68] who found that recognition accuracy was a more important predictor of acceptability than available vocabulary for a data entry speech system. Improvement in human-machine interaction may reduce the adverse effect of incorrectly identified commands on user satisfaction. One example of an improvement to human-machine interaction may be for the user to be informed of the reason for a problem. Boyce and Gorin [54, p.4] found that over a third of study participants who were prompted to retry a command, assumed that the problem was the understanding of words spoken (voice recognition error) rather than the command phrase not being understood (user knowledge/recall error coupled with low system habitability). Where a user expresses a command using a phrase which is not understood by the command recognition system, the process in Figure 3.5 shows a potential consequence of low habitability where interaction between a user and a voice-interface system could enter and remain in loop A until either the system takes an incorrect action or the user abandons the voice interface system and either forgoes the action or reverts to a conventional interface. An example of a possible improvement would be to inform user of the type of problem with the command, allowing the user to correct the input accordingly. For example, stating that a command was not understood due to ambient noise may not reduce perceived system capability (as the issue also occurs in human-human interaction). A user might then resolve such an issue by waiting for a moment when a masking noise is lessened or by speaking more loudly. If the reason given for the error were that the command wording was not understood, a user might elect to rephrase a command rather than speaking the same command more loudly, this may also providing an opportunity to catalogue the original request syntax to improve the command syntax corpus. Where a more habitable natural language method could be employed which allowed users to express a command using any command phrase, the only possible error-state would be that the command words were not correctly recognised.

Unanimous dislike for a voice command interface to take an incorrect action suggests that the threshold for deciding between acting or confirming an action may be an important human factors issue. This is supported by the result for question 45 which directly asked for a qualitative confidence threshold for taking an action. For this question, all but one of the participants stated the need for a high confidence threshold before an action is taken. It is not clear to what degree this is due to an inherent requirement that the system does not take an incorrect action or the degree to which participants had experienced difficulty with correcting incorrect actions.

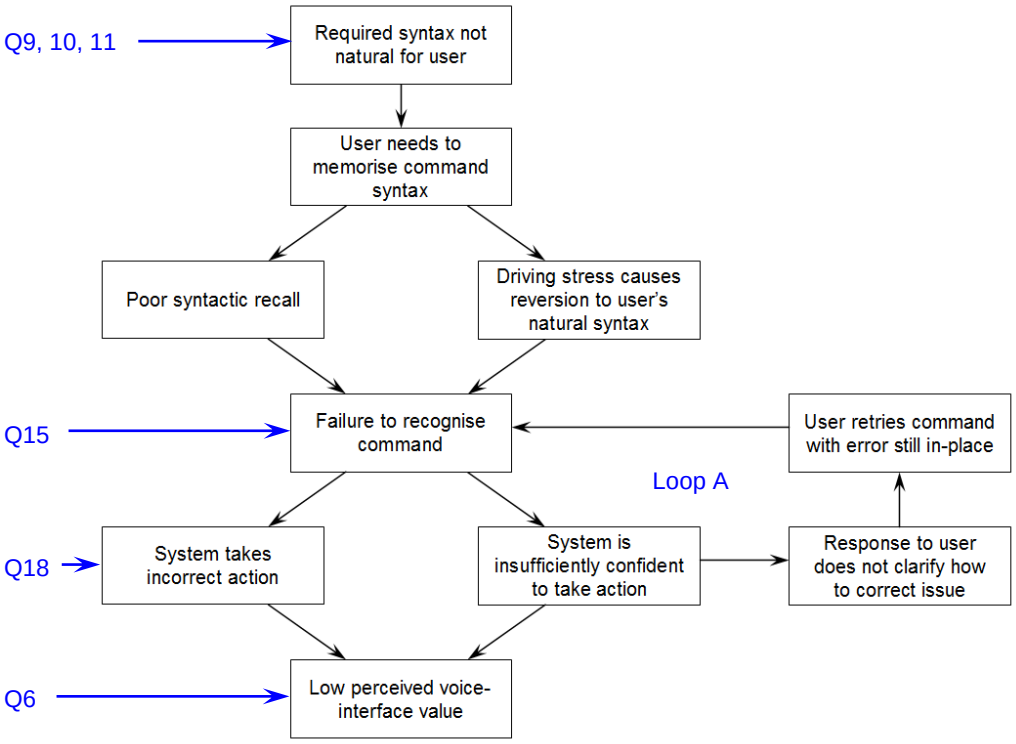



Figure 3.5. Possible process flow resulting from low habitability thereby causing low perceived value of voice interfaces (study 1).

Two potential situations where a dialogue between a user and a voice interface system may occur are to manage a situation where a command is not recognised (or pre-empt this by using a dialogue-led interaction in all cases) or alternatively dialogue may be used to elicit additional information where a command is ambiguous. Since the latter is implicitly required and is the case in human-human conversations, question 32 addressed only the use of dialogue to manage misunderstood commands. A majority of participants indicated a preference for repeating a command rather than entering into a dialogue. Possible reasons for this may include expectation of further recognition difficulty during the dialogue, concerns over cognitive load (needing to concentrate on the dialogue) or time to enact a command in this way.

Two factors relating to participants' experience constrained the topics examined by this study and these are possible areas for future research:

- As participants were only able to respond based on the existing voice interfaces of which they had experience, it was not possible to assess how the relationship between command identification and user satisfaction might be affected by improving the human-machine interface.
- All commercial voice-interface systems used by participants had an alternative conventional interface for all available actions. It was therefore not possible to assess how responses might differ if this were not the case.

## 3.7 Conclusions

Approximately half of the study participants were dissatisfied with commercially available voice-command user interfaces. Factors which affected this included:

1. Low habitability, i.e. the need for users to express commands using a pre-defined command phrase.
2. The proportion of times that a voice-interface system failed to immediately enact a command.
3. Any occasion where an incorrect action is taken by a voice-interface system.

User satisfaction with voice-interfaces, likelihood of employing such interfaces in preference to conventional interfaces and the opportunity to therefore realise the benefits of voice-interfaces may be improved by further research into methods which allow users to employ their own choice of command phrase and which allow a higher proportion of correct responses to spoken commands. In addition, developing methods to better identify and optimise a threshold for enacting a command may improve user satisfaction.

# *Chapter 4: Exploring the potential for crowdsourcing command samples using a typed modality.*

## Aim of Chapter 4

Humans are able to draw on a world view and an understanding of language when deciding the intent of a natural language command. Computers similarly require some form of knowledge structure which can be used to determine the intent of natural language commands. A data-set which might be used for this purpose is a corpus of commands previously given by humans. Spruit [14] shows that command syntax will vary by user's geographical origin which may necessitate samples from many locations. Crowdsourcing would allow many geographically distributed samples to be gathered without infeasible scaling in resource cost. Chapter 4 contains two studies which considers methods of gathering samples of command phrases (research question 1, section 1.4) and evaluating of the usefulness of the sample-data thereby generated (research question 2, section 1.4).

## Introduction

Study 1 (chapter 3) noted that user-satisfaction with voice interface is related to the need for users to learn fixed commands. Allowing voice interface users to employ any command phrase to enact a desired action would avoid the need for users to memorise fixed command phrasing. In section 2.5, it was noted that syntactic habitability (a subset of full habitability requiring only the interpretation of command paraphrases which a user might actually speak) is the focus of this thesis.

One method of enabling a voice interface to interpret any command phrases chosen by users could involve a comprehensive fixed phrasing method which contains patterns for all types of command phrasing which a user may employ. However, where a set of commands (for each action) intended to encompass those likely to be employed by users is chosen by system designers, the consequence is that 'natural language' becomes limited to either word patterns which are natural to the designers or which the designers perceives that other people might use. Since there is a large variety of dialects (and languages), the commands selected by designers may not meet the needs of all potential users. Gould notes that consulting users to identify their needs is necessary and that the consequences of designers failing to consult users include systems being "hard to learn", with "arbitrary inconsistencies with other systems" and that such systems can lack "insight into what users could really benefit from" [39, p.96]. One limitation of a comprehensive fixed command phrase approach is the potentially large range of semantically valid command phrasing which users may employ. Baber et al [7, p.50] notes variation in choice of words during an ATM-based study. Another limitation of a comprehensive fixed command phrase approach is that commands will not be recognised should any spoken words not be recognised by the voice recognition system and the method is therefore not robust.

Pattern-matching is an alternative approach to comprehensive fixed-syntax, this method examines commands for words or sentence fragments in order to identify an intended action. Pattern matching may be affected by the degree to which the same words and sub-sentence fragments are used in different patterns to achieve different ends and may therefore require increasingly complex rules as the quantity of possible actions (and therefore corpus command phrases) increases. Semantically valid commands only need to be pattern matched if users will employ them; this is noted by Watt [12] as semantic habitability. Where can I fill up my tank' is an example of a command which might be spoken by a user. 'How might I replenish my liquid petroleum' is an example of a command which whilst part of a fully habitable set of paraphrases (as defined by Baber [5]), might be omitted without any impact on a population of users. One way to ensure that pattern-matching rules are based only on command phrases which users will actually employ would be to obtain command samples from a sufficient quantity of people for each language and dialect. Use of command samples is effectively an empirical approach to excluding command paraphrases which fall within the definition of fully habitable but outside the definition of fully syntactically habitable. Removing command paraphrases which no user will employ reduces the complexity of the command processing task and using example commands to achieve this precludes designers having to try and estimate which command phrases can be excluded.

One approach to obtain command syntax from a large set of people is to use existing natural language material (such as literature, radio recordings etc). Machine learning systems can use existing human-language material to build semantic probability structures which assign the meanings of words and the likelihood of phrases and word sequences, with the goal of understanding the intent of a wide range of possible phrases. However, speech patterns may be affected by the context in which they are generated. Schmidt et. al. [69, p.9] show that context-awareness can improve interfaces on mobile devices and Baber and Noyes [8, p.149] note that "With speaker-dependent devices, it is necessary to ensure consistency in speech patterns between enrolment and use". A driving context can generate user-stress which may differ from the context in which existing natural language material might be generated. The potential for stress to affect speech is noted by Baber and Noyes [8, p.149] who summarise material suggesting that speech patterns are stress and therefore context related. Also, human-to-machine may differ from human-to-human language as shown by Dahlbick and Onsson [37, p.291] who noted that "the language used when communicating with a real or simulated natural language interface" was different to human to human conversation.

An option for obtaining commands within a context similar to that in which they will be used and without incurring costs which scale with sample quantity would be to record a set of users interacting with a natural language speech system whilst driving on public roads. A natural language system used for this could not be fixed, guided or limited to a command subset which

might affect selection of command phrasing by a study participant. This may preclude current commercially available automotive systems. Gathering samples using this method may therefore need to be done either by a controlled study or by first releasing a commercial natural language system. In the latter case, a commercial system released for this purpose would need to meet the performance thresholds identified in section 3.5. If this were not achieved, this could result in users moderating their language in an attempt to improve performance, thereby making the resulting command samples differ from users' natural speech. In addition to this, commercial companies may be reluctant to release natural language voice interfaces which demonstrably do not meet performance thresholds which are acceptable to potential customers.

Samples of user commands obtained via controlled studies may need to originate from people whose choice of command phrases is similar to that of a user. To facilitate this for all possible users, the quantity of samples needed is affected by several factors including the geographic area over which the method is to be deployed, the dialect and language types over this area, the degree of language variation with geographic distance and how closely command samples need to match a user's choice of command phrase. Chan and Khalid [70, p.1388] describe the high degree of language variation in several countries and Spruit [14, p.10] shows how syntactic dialect varies with geographic distance in a study performed in Holland. Since recording live usage of automotive voice interfaces may not be possible, obtaining samples via controlled studies is an alternative option, however such studies will need to be cost-viable for the potentially large number of samples and distributed study participants. Options for studies to obtain command samples include:

1. Recording command samples during live vehicle driving using a 'Wizard of Oz' (WOz) method (which uses a concealed human operator to simulate a natural language voice interface).
2. A WOz method using a vehicle driving simulator
3. Voice recording of study participants by researchers
4. Voice recording of study participants using participants' own computer equipment

Both WOz methods require equipment and human resource which scale with number of samples and may not be viable for broad geographic studies. Voice recording by researchers, for example using street interviews do not require complex equipment and would potentially be less expensive than vehicle or simulator studies, however human resource would still be proportional to the number of samples required and would similarly scale with geographic region. Also, it may be difficult to establish a sufficiently controlled condition for this approach. Zoltan-Ford [18, p.529] explains how hearing spoken words may condition subsequent spoken syntax and a method would be needed of prompting participants without inducing such a conditioning effect. Other study factors may affect this method (examples

include the presence of a researcher and consequent time-pressure, potentially a public setting) and these factors may also affect command phrasing.

Option 4 involves obtaining the required information by crowdsourcing which eliminates the issue of scaling the human resource needed for running studies. It would be ideal to obtain command samples using the same modality as that employed during live use (i.e. a spoken modality) and as part of this thesis, an attempt was made to construct a method of recording study participant's spoken commands using a purpose-designed software application. A preliminary trial of this method worked as intended in some cases and a number of successful study responses were obtained. However, a range of issues occurred in other cases which prevented completion of the study. Examples were:

- Computer microphone not fitted, not turned on or the computer does not have a working software driver.
- Application prevented from running by security software or requires user override of a visual security warning.
- Poor or indistinct commands

Results of the study demonstrated the potential for data gathering by crowd-sourcing using participants own computer equipment; however it also demonstrated that speech was not a common input modality for a home computer and that this therefore presented a number of issues which were not easily addressable. It was also observed that human transcription to text would still be needed to avoid speech recognition issues and this would require human language interpreters and costs which again scale with geographic region.

An alternative method to crowdsource commands is to employ a typed modality as this is standard for almost all home computers. For typed modality command samples to be an acceptable basis for a natural language corpus, it would be necessary to demonstrate that any difference in wording of commands between spoken and typed modalities would allow (for the chosen method of natural language command processing) intended actions to be identified with any reduction in probability of correct action identification being sufficiently small that results remained above the user-acceptable threshold or where any reduction in performance could be overcome by viable methods (e.g. increasing the corpus sample count by a manageable quantity). The two studies in section 4 consider the differences between typed and spoken modality command samples.

Section 4.1 describes study 2, the first of two studies in this section in which paired spoken and typed modality command samples were compared. Common prompts were used to elicit commands to evaluate the ways in which command phrases differed and to consider the degree to which this might affect usage of typed modality samples as a corpus for a voice-interface natural language interpreter.

In section 4.2, study 3 extends the evaluation of modality effect to an automotive context, developing the prompting mechanism to shows animations of action outcomes and thereby further minimise preconditioning of participant responses.

# 4.1 Preliminary user study on crowdsourcing and typed modality commands (Study 2)

## 4.1.1 Introduction

### Aim of Study 2

Obtaining sample natural-language commands from study participants requires a method of prompting participants as to the action that they need to take and a method by which they can express and record commands. Study 2 investigates the possibility of using a static image as a prompt and assesses the use of a typed modality to facilitate communication of commands.

### Overview

Collecting a large and geographically distributed set of command samples could be expensive and resource intensive. These issues may be overcome if it was possible to collect samples by crowd-sourcing via the internet, social media or email as this would allow many participants to provide responses simultaneously and with no human assistance. Applications able to record sound and/or identify spoken words are not in common use in home computing & requiring participants to add this capability to home computing equipment could provide a barrier to participation. Spoken commands would then require transcription into text for use by a natural language processor, incurring a resource cost for human transcription. Use of voice recognition engenders the potential for errors in transcription.

Using a typed modality would overcome these issues. It is however possible that the syntax of typed commands may differ from spoken commands. The intent of study 2 is to obtain two comparable command-sets in typed and spoken modalities to investigate potential differences in command phrasing.

Spoken language is a common modality for human interaction whereas typing is a skill acquired based on need. Typing competence may therefore vary across the group of participants. Differences in command phrasing may increase with differing levels of experience with the two modalities. To assess this, participants' typing speed was measured to examine potential correlation with differences in command phrasing.

When speaking a command to a researcher, study participants responded with minimal hesitation and did not have the option to review and edit their wording. When providing typed commands, participants had the option to consider and edit their command without the time pressure of a waiting researcher. Two questions were put to participants to assess whether thinking time and the opportunity for self-editing may affect choice of command phrasing.

Reading words may condition study participants' subsequent choice of command phrasing. The method by which a desired action is communicated to study participants would ideally

avoid the use of any text. This study assesses one method of graphically indicating a desired intent.

## 4.1.2 Method

### 4.1.2.1 Study design

Sixteen study participants were individually asked to give command wording for twelve pictorially prompted actions. Command samples were obtained from each participant on two occasions, once in a spoken modality and once in a typed modality. Eight of the study participants gave their spoken responses first and typed responses second. The remaining eight were questioned in the reverse order to counter-balance the study. The two sets of responses were separated by a period of four weeks to minimise the risk of that a participant responding during the second stage might remember their previous answer. Also, that there would be a second stage to the study was not revealed to participants until the four week period had passed.

Whilst this thesis addresses an automotive domain, the systems addressed by prompted actions used during this study lay outside of this domain. This decision was based on the potential noise-factor of variation in study participant exposure to existing automotive voice interfaces. To avoid this, commonly in-use systems were selected (that is, systems which all study participants would have experience using) but which did not (at the time) have a voice interface and for which participants could not therefore vary in voice interface usage experience.

### 4.1.2.2 Procedure

In the case of the typed modality, participants were provided by email with a copy of the survey formatted as a Powerpoint document. The email informed participants that the survey would take approximately ten minutes to complete and that they should do so when they would not be disturbed. The only environmental stipulation was that the participant should be alone and not under time pressure. Participants returned the finished survey by email using a link provided on the final page. Each page within the survey contained written instructions on how to complete the questions and the participant was advised on the email to take note of these. For each command, a numbered box was provided into which the participant typed their response.

The spoken modality survey was conducted as a structured interview. Participants were given guidance using the same words that were stated on the typed-modality sheet and shown the same images on paper. If a participant requested clarification on any of the information or visual prompts, the advice was either repeated or participants were advised to respond as they thought best. Each participant stated their response verbally and the researcher recorded this by hand.

In addition to the commands, the typed modality part of the study gathered three additional measures, two overt and one concealed. The overt measures were taken via sliders on the final page of the sheet. Participants were asked to decide firstly between whether they generally pause to think or respond to a question on instinct and secondly between whether they prefer to communicate via email or telephone. The concealed measure was typing speed; the spreadsheet recorded the duration between each key-press while participants were typing commands.

#### 4.1.2.3 Participants

Sixteen participants were selected with ages ranging between 20-60 years and from both genders. All of the participants lived in the Warwickshire area within the United Kingdom and were drivers.

For the typed-modality part of the experiment, it was necessary that all participants had access to a computer and a copy of Microsoft PowerPoint. No participant was excluded as a result of this requirement.

#### 4.1.2.4 Selected actions

Twelve actions were selected across three commonly used electronic systems which were not (known at the time to be) operable using voice commands. These included a mobile telephone, a television set and a vending machine. Automotive actions were not included to avoid the potential that participants had different levels of exposure to automotive voice interfaces. Actions were selected to elicit a range of word-counts. The actions comprised:

Mobile telephone:

- Turning off the phone
- Opening the menu
- Selecting number one
- Using the loudspeaker for a call
- Answering an incoming call
- Editing the address book entry for ‘Andrew Jones’
- Calling ‘Andrew Jones’

Television set:

- Setting volume to mute
- Pressing the red button
- Selecting channel one

Vending machine

- Selecting item 44
- Returning change

#### 4.1.2.5 Prompt method

For each action, participants were shown an image depicting a conventional button interface with a numbered arrow identifying a specific button. A text description explained that the participant was to consider (given the visual state of the system) the effect that pressing the button would have and, assuming that a fully capable voice-interface was to be used instead, state a command. The first command prompt is shown in Figure 4.1 and the full set is shown in appendix A.


Figure 4.1. Pictorial prompt for the action of turning off a mobile telephone (study 2).

### 4.1.3 Results

The degree to which a command processing system may be able to use a typed command corpus to identify the action intended by a spoken command may relate to the commonality of the typed and spoken command wording (i.e. the lexical similarity). For all participants, 316 words were spoken and 348 words were typed. The set of spoken commands had an average word count of 1.65 and the set of typed words had an average word count of 1.81. The set of spoken commands and typed commands were assessed for significance of word-count difference using a paired t-test which showed typed word count as significantly greater [t (191) = 2.59, p=0.01]. The possibility that the additional typed words were politeness prefixes or suffixes was eliminated as this was noted in only one case.

The words spoken and typed were compared using three sub-groupings as shown in Figure 4.2:

Subgroup A: Each individual participant's response for a specific action (Figure 4.3)
Subgroup B: For each action, the entire group of participant commands (Figure 4.4)
Subgroup C: For each participant, commands for the entire group of actions (Figure 4.5)

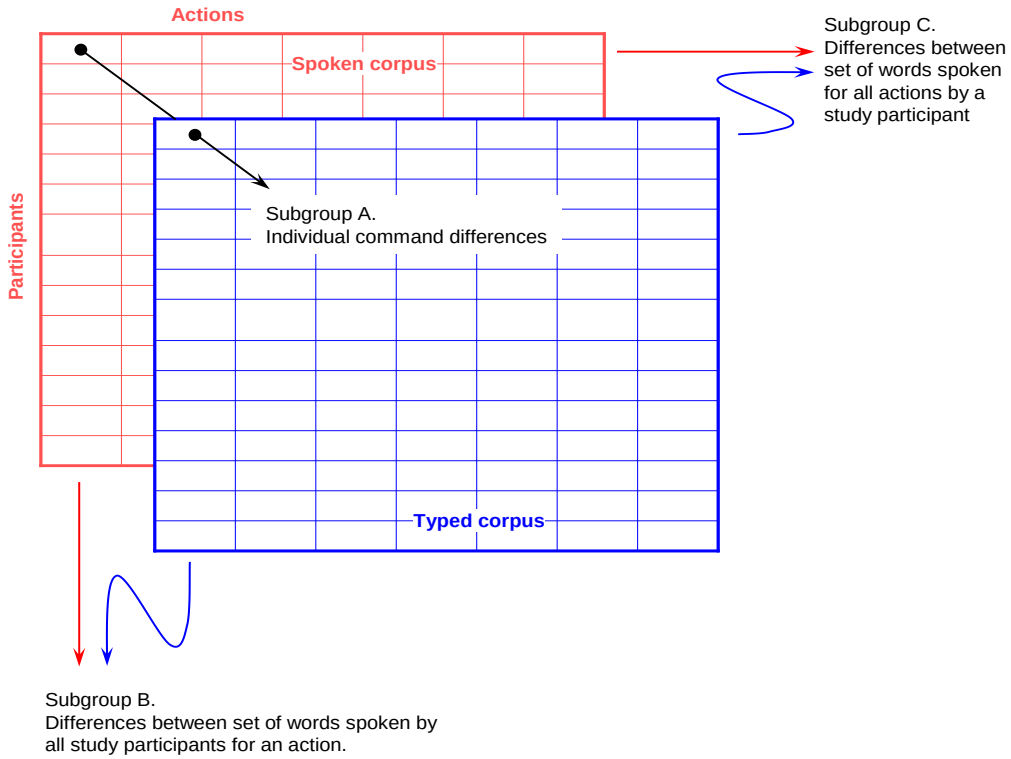



Figure 4.2. Rational sub-grouping of command sample corpus for modality comparison (study 2).

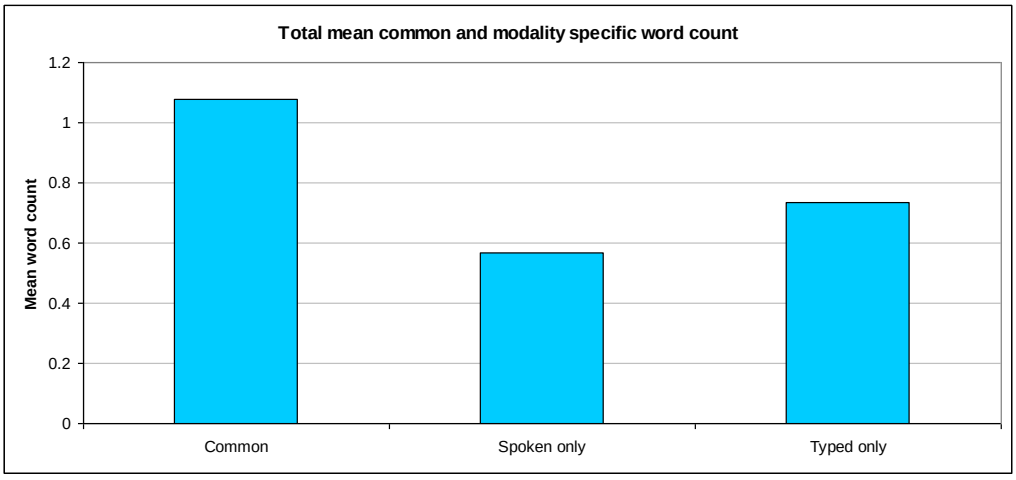



Figure 4.3. Mean corpus word count by rational subgroup A: individual participants and actions (study 2).

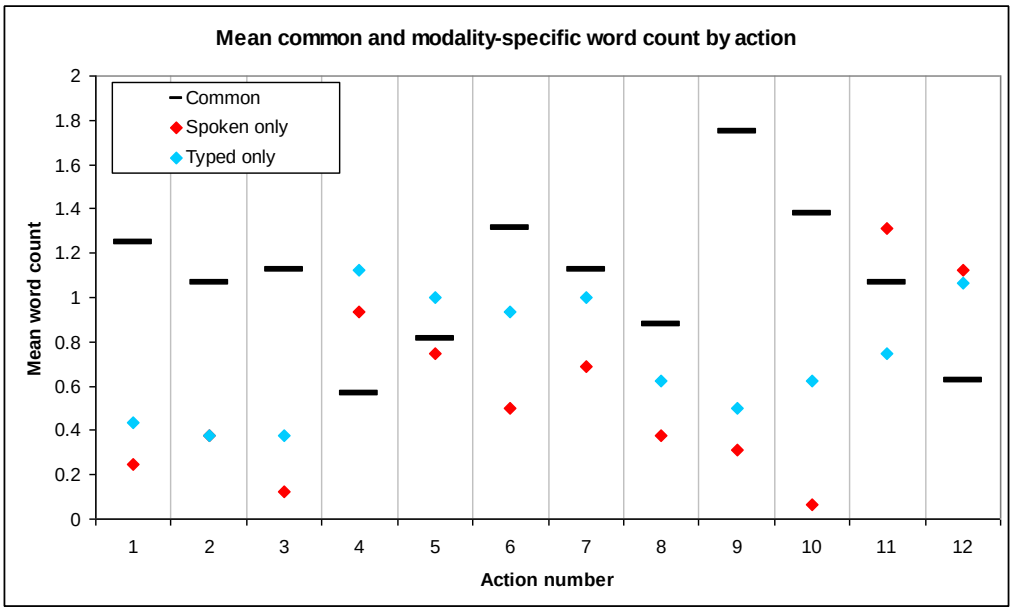



Figure 4.4. Mean corpus word count by rational subgroup A: grouped by action (study 2).

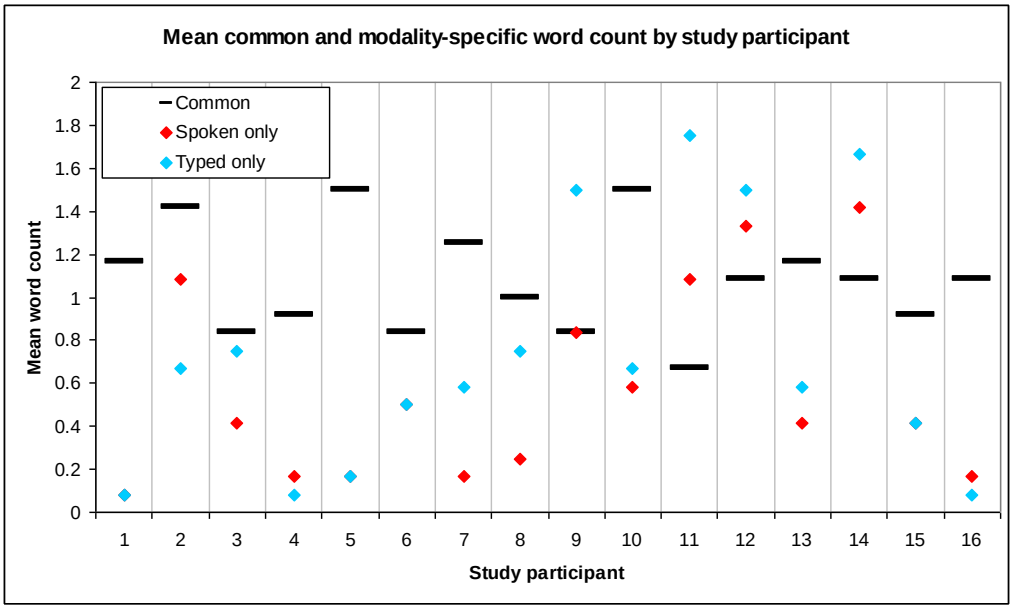



Figure 4.5. Mean corpus word count by rational subgroup C: grouped by participant (study 2).

The proportion of command words which were both spoken and typed (compared to those unique to either modality) may increase as the average number of word in a command increases. Figure 4.6 shows the correlation (including a 95% confidence band) between average count of command words and proportion of command words which were both spoken and typed [$F(1,190) = 118.5$, $p<0.001$] with an $R^2$ of .384. This suggests that the effect of using a typed modality corpus to identify spoken commands may decrease as command complexity increases.

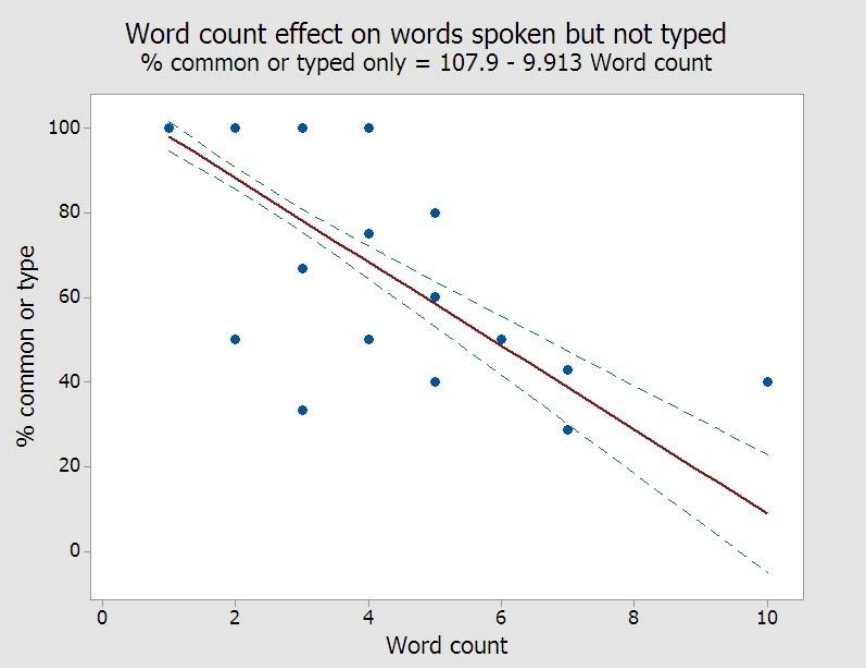



Figure 4.6. Sample word-count effect on commonality of typed and spoken modality commands (study 2).

Participant-related factors which may contribute to the relationship shown on Figure 4.6 were assessed using subgroup C (by participant) and include participant typing speed, experience with a typing modality and likelihood of self-editing (i.e. re-thinking) a typed response.

To assess effects of participant typing speed, the time gap between each key-press was recorded for each participant during the study. The resulting distribution is shown as median and inter-quartile ranges (Figure 4.7). No correlation was found between median typing speeds and proportion of command words which were both spoken and typed (Figure 4.8).

Participants were asked to indicate their preference between communicating via email or telephone call to assess their possible experience with a typing modality. This was compared with the proportion of spoken words also in typed data. The results are shown in Figure 4.9. The correlation was not statistically significant.

Likelihood of self-editing was assessed by asking participants whether they generally pause to think before responding or respond more immediately. This was compared with the proportion of spoken words also in typed data using subgroup A and a correlation was found [$F(1,14) = 4.85$, $p=.044$] with an $R^2$ of .258. The results are shown in Figure 4.10.

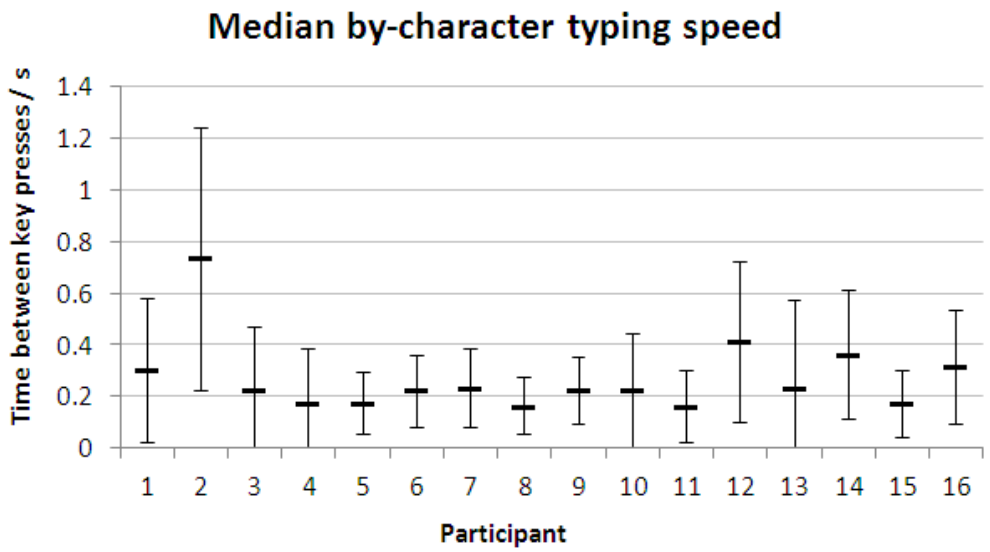



Figure 4.7. Median and inter-quartile range for participant typing speed (study 2).

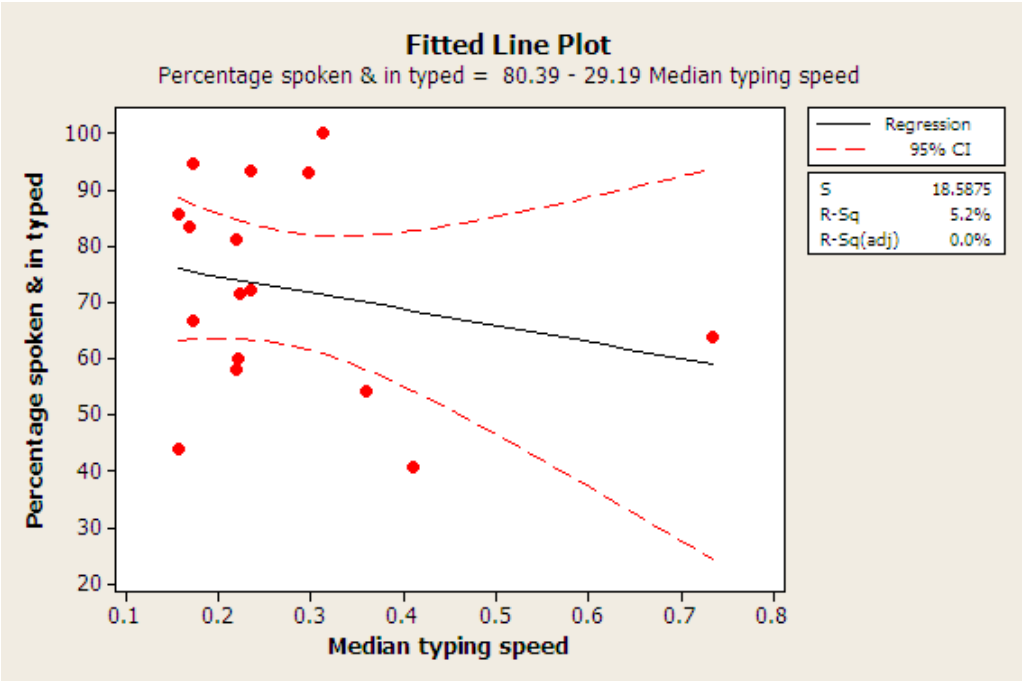



Figure 4.8. Correlation between mean typing speed and commonality of spoken and typed modality words (study 2).

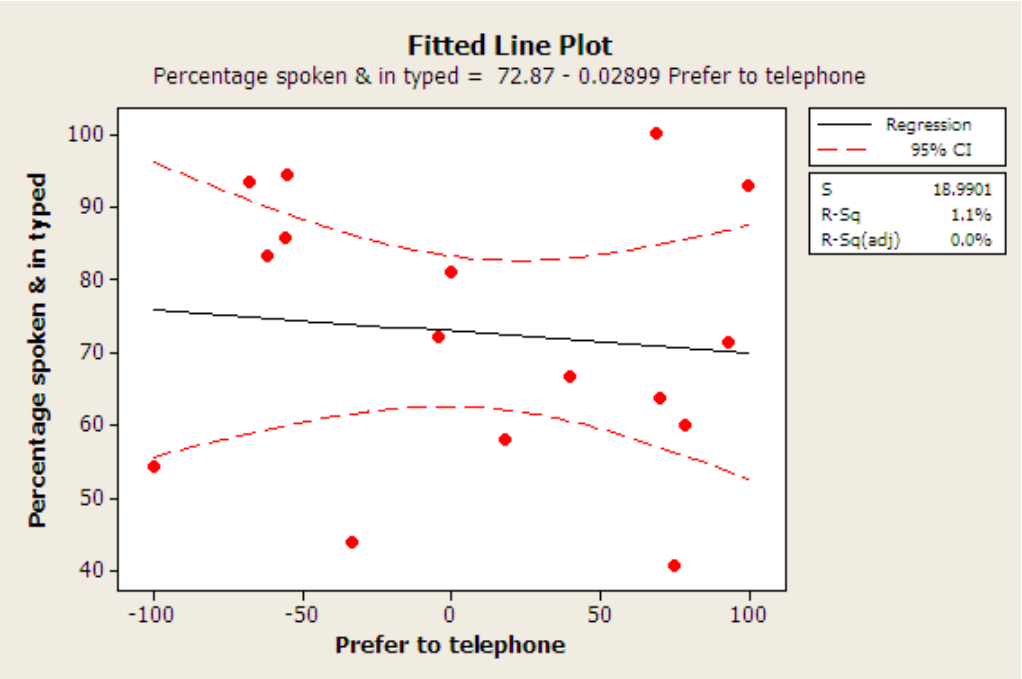



Figure 4.9. Correlation between preference for spoken human-human communication and commonality of spoken and typed modality words (study 2).

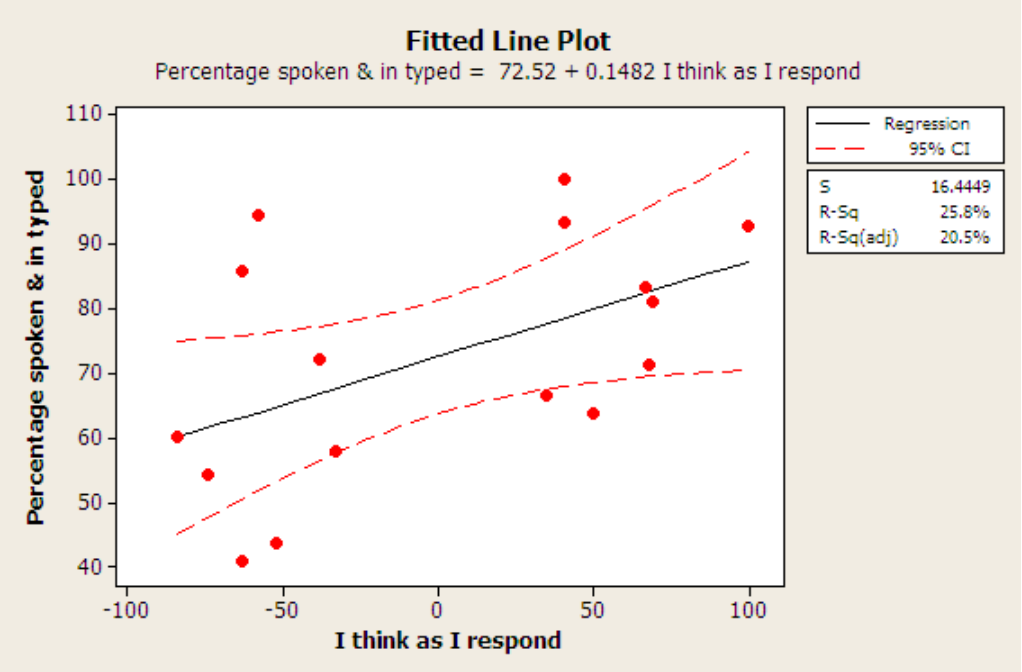



Figure 4.10. Correlation between preference for reactive communication and commonality of spoken and typed modality words (study 2).

Prompts presented to participants showed conventional control interfaces and the instruction provided required commands to be given as a surrogate for these controls. When considering these results, it should be noted that many of the participants did not follow instructions and instead gave commands to operate the conventional control. For example, if the context were a television remote control and the numeric keypad button '1' were indicated, it might have

been expected that a participant would give a command to 'select channel one', with the television state as the context. During the study, many participants gave commands such as 'press button one', selecting the visually prompted remote control handset as the context.

### 4.1.4 Discussion

#### 4.1.4.1 Modality differences

Typed commands had a significantly higher word-count than spoken commands and the two modalities must therefore result in command phrasing which originates from different populations. One possible explanation for this difference is that words from spoken commands comprise a subset of their typed equivalents. However, some spoken commands contained unique words, suggesting that there must be other factors which cause differences in command phrasing.

Figures 4.4 and 4.5 showed differing degrees of common and modality-specific word content. Actions were shown in Figure 4.4 to differ considerably in the proportion of spoken-unique words and this suggests that actions may have variable susceptibility to modality effect on action identification. Participants also differ greatly (Figure 4.5) in the same regard; this suggests that modality effect on action-identification will vary between individual users.

#### 4.1.4.2 Factors affecting modality differences

Three factors were measured and compared to the percentage of spoken command words which were also in the corresponding typed command:

Familiarity: One hypothesis was that typing speed may relate to a participant's degree of familiarity with the typing modality and therefore may relate to similarity of typed and spoken command phrasing. This was not supported by the results.

Preference: The second measure considered whether participants preferred to communicate by telephone or email. This assessed whether modality preference related to similarity of typed and spoken command phrasing. This was also not supported by the results.

Self-editing: Whether participants pause to think before responding to questions or respond immediately was measured to assess the potential for self-editing (either by thought or visual editing) typed responses. This showed a significant effect on the similarity of typed and spoken command phrasing.

If pausing to think before responding does relate to potential for typed command editing, one possibility to improve similarity of spoken and typed modality command phrasing may be to design a typed modality data gathering method to induce a response time-pressure. This may reduce the opportunity for self-editing by thought. Another option could be to prevent changes after a command has been entered.

#### 4.1.4.3 Optimising prompt methodology

The prompting method used in this study was selected to avoid two potential problems:

1. Commands prompts could have conditioned participants choice of command phrase.
2. Prompts could have been unclear or indicate more than one potential action.

To prevent conditioning of participant selection of command words, prompts were graphical with no text content. To allow users to understand the intended action, the prompts showed a commonly used conventional control (and where needed, a system state) with the intent that using the conventional control would result in only one possible action. The results did not indicate that any conditioning of commands occurred and this was therefore successful. However, in order to allow this method to be used, the twelve actions were carefully selected. Any command which might include user-specific data could not easily be prompted using this method, for example instructing a music system to use a play list would be difficult to show graphically without including some descriptive words. Whilst the method used in this study had no apparent conditioning effect, it may therefore be limited in application.

Whilst results did not indicate that participants found systems and intents unclear, an unexpected result of the prompting method was that command were given which instructed operation of the conventional interface, not to operate the intended system. For example, one participant stated ‘press the red button’. Reasons why this may have occurred include:

1. Some participants may have been visually oriented and simply addressed what they saw
2. Participants needed to identify the resultant effect of pressing the indicated button and then reverse the thought process to consider a verbal command. This may have been too complex a request and participants simplified this by disregarding the effect stage.
3. The instructions were insufficiently clear
4. Visual prompts and pre-experimental instructions seemed to differ in intent and the visual prompts were more immediate and clearer.
5. Participants were accustomed to thinking of the intended system via the conventional interface and when asked to operating the intended system, did so via the conventional control.

As some participants gave commands addressing the intended action and some gave commands with the conventional control as the context, the results of this study may contain two or more differing populations of results, potentially resulting in bimodal or multimodal data.

Improving the prompting method may therefore result in a single data population, improved sensitivity in understanding modality effect and also a more effective corpus for natural language processing.

#### 4.1.4.4 Command length

Actions were selected to generate more than two words per command. The intent was that commands given by participants would be in full sentences or at minimum, that two words would be required to express an intent, similar to the simple verb-noun pairs used in Scott Adam's interactive fiction games (noted by Antoniazzi [71, p.70]). However results showed an average of less than two words per command. Baber [34, unformatted] notes terse human speech to machines, observing that "subjects' speech to the computer was short, succinct and task specific; using simple imperative commands and used a restricted vocabulary" and that "subjects' speech to the computer was very different to the speech they used to other people". A potential reason for this is that the degree of context provided may have resulted in participants assuming that only one action would be possible for a function (given the apparent system state) and that stating the function would therefore be sufficient. An example where a verb was implied was the prompt to use the mobile telephone's loudspeaker to engage in a hands-free or group conversation. In this case, five of the participants simply said the word 'loudspeaker'. Since this was a 'hot' button on the conventional control (the effect of pressing the button was shown on the screen), participants may have assumed that as the option was presented, the loudspeaker could not have already been on and that turning it on was therefore the only valid verb. Another possible reason for the low word-count is that participants may have perceived that a verb had only one valid function and that the verb alone would be sufficient. An example of an implied function was the action to turn off the mobile telephone. For this action, seven of the participants simply stated 'off' as the command. Since no other function was apparent which might be turned off, participants may have assumed that specifically directing the command to the mobile telephone operating system was unnecessary. It is also possible that the low word-count resulted from users attempting to simplify command syntax with the expectation that this would help the computer interpreter to correctly identify their intent.

Selection of actions which in order to be unambiguous require a command containing more words may improve this. This may require improvement in the method of prompting so that a complex action can be communicated to a study participant whilst avoiding preconditioning participant choice of command wording.

### 4.1.5 Conclusions

Figure 3.3 shows the degree to which user acceptance of voice-interface systems is related to the accuracy of action identification. Results from this study have shown differences between the phrasing of spoken and typed modality commands that may adversely affect spoken command identification when using a typed modality. Determining the magnitude of this effect would require a command interpreter which makes use of such data.

A number of factors could be improved in a further study which may allow the potential effect of using a typed modality corpus to interpret spoken commands to be considered in more detail:

#### Prompting method

Obtaining a corpus of command samples by crowdsourcing may require a method of prompting study participants which avoids preconditioning whilst unambiguously communicating desired actions. In this study, the method of prompting avoided preconditioning (which required careful selection of the set of actions) but caused ambiguity when indicating the desired action. In addition, some participants addressed the conventional interface displayed rather than the system it controlled. One criterion was the need to indicate the action with a simple diagram and basing selection on this requirement may have resulted in low command word count. An ideal prompting method would allow any action to be communicated to study participants without causing preconditioning of command phrases and whilst ensuring that the subject of the command is unambiguous.

#### Context

The context for this study was a conventional control for a commonly in-use system for which voice control was not (known to be) commonly available. Whilst this allowed study of modality effect whilst avoiding variation in participant experience, material is cited in section 1.2 which shows that syntax may be related to command context and that understanding modality effect within an automotive context may require a study to also use this context.

#### Sample-count

Developing a corpus for natural language command processing may require understanding of the relationship between sample count and probability of correct action identification. This understanding may allow use of an increased sample-count to mitigate any effects of using a typed modality for the corpus.

Increasing the number of command samples may have diminishing returns in terms of command-identification. Understanding this relationship may be necessary as resource needed to gather samples may be a constraining factor. In addition, increasing sample-count

by including samples from a geographically larger area (i.e. samples which are less specific to the dialect of a user) may be affected by the regional speech variation found by Spruit [14] and this may become detrimental to action identification.

Study 3 (section 4.2) assessed a potential way to further develop the prompting method such that participants had a common view of the intended action. Study 3 also assesses commands given in response to actions which are selected to require a higher word-count to fully (unambiguously) state and which have an automotive context.

## 4.2 Detailed user study on crowdsourcing and typed modality commands (study 3)

### 4.2.1 Introduction

#### Aim of Study 3

Study 3 expands upon study 2 (section 4.1) by investigating the differences between commands given in typed and spoken modalities within a vehicle context, using a revised prompting method suitable for crowdsourcing. The first aim was to establish whether an animated prompting method which is focussed on the expected outcome improves upon the results of study 2 by engendering a common understanding of the required action for all participants. The second aim was to assess a vehicle command context and understand the effect of using a typed modality given a set of actions which engender commands which have a higher word-count.

#### Overview

Prompting actions by showing pictures of conventional interfaces resulted in some study 2 participants addressing commands to the conventional interface rather than the intended system. For example, a participant might say 'press button one' rather than 'please select station BBC1'. In study 3, the static image showing a way of issuing a command was replaced with an animation showing the desired effect of a command. The refined prompting method was also designed to minimise participant conditioning and to encourage participants to respond based on their initial thought of a suitable command irrespective of modality.

Study 2 found that participants gave many one-word commands, implying the noun or verb. This tendency may be explained by Amalberti et. al. [9, p.4] who summarised a range of material which suggested that users may simplify command syntax with the expectation that this will help the computer interpreter to correctly identify their intent. For example, 'please open the boot' might be expressed 'boot' if the user was only aware of one boot-related command.

In study 3, a set of actions were selected that required a varying number of words to fully identify the action prompted. For example, whereas a command to turn on the rear fog lights could be expressed as a single word 'fogs' (implying the verb in the context that the fog lights are currently off), lowering the front passenger side window half way requires a minimum of five words to fully describe. Some commands were selected to require additional data items to describe the type (e.g. up or down) or required degree of change (e.g. half-way).

Samples taken from the same participants in both spoken and typed modalities were considered from the perspective of the following dependent variables:-

- Command length (verbosity)
- The degree to which commands matched prompted actions (system/function ambiguity)

- The amount of required information omitted from commands (total information ambiguity)
- The increase in proportion of words common to both modalities as sample size increases (effect of sample-size).

The data gathered in study 3 is further evaluated in chapter 6 following the development of a natural language processor in chapter 5.

## 4.2.2 Method

### 4.2.2.1 Participants

Twenty participants were selected across a range of ages (20-60 yrs) and from both genders. All of the participants lived in the Warwickshire area within the United Kingdom and were regular drivers. For the typed-modality part of the experiment, it was necessary that all participants had access to a computer and a copy of Microsoft PowerPoint. No participant was excluded because of these requirements, demonstrating accessibility of the approach.

### 4.2.2.2 Selected actions

The twelve actions selected are shown in Table 4.1. They were chosen to include a range of overall system functions (e.g. ventilation, lighting), and to vary in complexity. In particular, some actions related to simple system requests with no potential ambiguity with other actions (e.g. action 1; to open the sunroof to a tilted state). By contrast, other actions were selected to require multiple requests (e.g. task 4) or included elements in common with other actions (e.g. actions 8 and 11 both refer to “wipers”).

Table 4.1. List of automotive actions prompted (study 3).

| Action | Description |
|---|---|
| 1 | Open the sunroof to a tilted state (i.e. not retracted) |
| 2 | Turn on the right indicator |
| 3 | Turn on the hazard flasher lights |
| 4 | Change dashboard display from odometer to 'trip' and reset trip distance (to zero). |
| 5 | Raise in-car temperature to 22 degrees centigrade and set the ventilation fan to full. |
| 6 | Tilt the steering wheel down. |
| 7 | Lower the front passenger window halfway down. |
| 8 | Set the front windscreen wipers to intermittent. |
| 9 | Increase the dashboard illumination. |
| 10 | Rear left headrest raise |
| 11 | Rear wiper on |
| 12 | Rear fog lights on |

### 4.2.2.3 Method of prompting

Visual animations were used to prompt participants to choose commands for each of the 12 actions. Once a participant had pressed a key to commence the animation, the screen showed a static line-drawing of a vehicle (in whole or part) which then animated to show a command being enacted. As an example, for action 1, the whole vehicle was shown in side view and the

sunroof then smoothly raised to a tilted position. Figure 4.11 shows the animation used for action 1 in its start and end-state and the full set of actions are shown in appendix B.

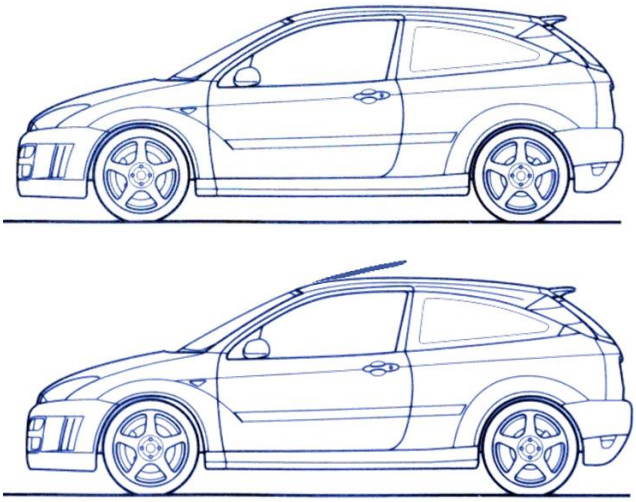


Figure 4.11. Appearance of the prompt for action 1 at the animation start and end (study 3).

Study two used a fixed image which was constantly visible to the participants which may have facilitated careful consideration prior to the typing of a command and editing once the command had been entered, something not possible in the spoken part of the study. In this study, the animation was transient and was removed when concluded and the participant then needed to consider both what action the animation was indicating and then decide upon a command phrase. Since this was a more demanding task and the animation was not present while this was being considered, the opportunity for a response considered at length may have been reduced.

#### 4.2.2.4 Study design

Command samples were gathered for the same set of actions twice for each participant, once for the spoken modality and once for the typed modality. The modality order was counter-balanced with half the participants addressing the typed modality first and half the spoken modality. A duration of four weeks was allowed between the two experiments to reduce the potential for cross contamination of the same command in different modalities.

#### 4.2.2.5 Procedure

Having given written consent to take part, the study was presented to participants in the form of a Microsoft PowerPoint file. In the case of the typed condition, participants were sent the file by email. The email instructed the participant to proceed when they had a sufficient time to complete the study and would not be disturbed. A text box was added below each of the animations within the Powerpoint file where participants could type their commands. Upon completion of the typed study, participants followed instructions on the final page to save and

return the results by email. For the spoken part of the study, the researcher was present and the Powerpoint file was presented on a laptop. Verbal commands were recorded using a mobile telephone application for later translation. The researcher gave no information during the data collection.

The study sequence comprised an instruction page and then twelve pages each of which played a single visual animation showing a vehicle action taking place, for example the sunroof raising. A green square appeared in the bottom right corner of the page at the conclusion of the animation to indicate to the participant that the animation was complete.

## 4.2.3 Results

### 4.2.3.1 Verbosity

The total number of words for each participant was summed for spoken and typed modalities resulting in the data shown in Table 4.2.

Table 4.2. Total number of words used for typed and spoken commands (study 3).

| | Sum of words by participant | |
|---|---|---|
| Participant | Spoken | Typed |
| 1 | 29 | 35 |
| 2 | 33 | 46 |
| 3 | 40 | 39 |
| 4 | 33 | 41 |
| 5 | 37 | 38 |
| 6 | 35 | 37 |
| 7 | 36 | 33 |
| 8 | 41 | 58 |
| 9 | 36 | 35 |
| 10 | 35 | 39 |
| 11 | 37 | 39 |
| 12 | 44 | 45 |
| 13 | 22 | 33 |
| 14 | 52 | 45 |
| 15 | 31 | 35 |
| 16 | 28 | 32 |
| 17 | 44 | 50 |
| 18 | 43 | 46 |
| 19 | 34 | 36 |
| 20 | 42 | 48 |
| Average | 36.6 | 40.5 |
| Average / command | 3.05 | 3.375 |

A paired t-test was performed on the spoken and typed data and it was found that typed commands had a significantly greater word-count than spoken commands [$t(19) = 3.16$, $p = .005$].

### 4.2.3.2 System/function Ambiguity

Ambiguous commands are those which do not fully define the intended system and function for the required action. Table 4.3 shows the total number of ambiguous commands made by each participant for the typed and spoken modalities.

Table 4.3. Level of ambiguity for all actions by participant and modality

| | Sum of coded System/function ambiguous commands | |
|---|---|---|
| Participant | Spoken | Typed |
| 1 | 6 | 1 |
| 2 | 3 | 0 |
| 3 | 0 | 0 |
| 4 | 3 | 2 |
| 5 | 7 | 5 |
| 6 | 3 | 1 |
| 7 | 2 | 4 |
| 8 | 3 | 0 |
| 9 | 1 | 3 |
| 10 | 1 | 0 |
| 11 | 0 | 0 |
| 12 | 3 | 1 |
| 13 | 6 | 2 |
| 14 | 3 | 3 |
| 15 | 2 | 2 |
| 16 | 3 | 4 |
| 17 | 5 | 2 |
| 18 | 3 | 2 |
| 19 | 6 | 5 |
| 20 | 3 | 2 |
| MEAN (Number and %) | 3.15 (26%) | 1.95 (16%) |
| Standard deviation | 1.98 | 1.64 |

A paired t-test was performed to evaluate if the two modalities had significantly different levels of system/function ambiguity. This found that there was a higher number of unambiguous commands given for the typed versus the spoken modality [$t(19) = 2.90$, $p = .009$].

### 4.2.3.3 Data Ambiguity

As well as the subject (noun) and action (verb), some commands required additional data. Examples include selecting from multiple instances (such as ‘front’ seats), choosing between categorical states (e.g. headlights ‘dipped’) and indicating a desired level (e.g. temperature ’24’ degrees). Results from the two modalities were compared to assess whether omission of required data occurred at a similar level for each modality. Action 7 (to lower the front passenger window halfway down) was selected to assess data ambiguity as it required three discrete items of data for the command to be fully detailed. The items of data comprised the front of the vehicle, the left hand side of the vehicle and that the window should be lowered half-way. The number of data items provided by participants for Action 7 was counted for both modalities. Table 4.4 shows the results.

Table 4.4. Count of data items missing for action seven by participant and modality (study 3).

| | Data items missing for action 7 | |
|---|---|---|
| Participant | Spoken | Typed |
| 1 | 2 | 0 |
| 2 | 1 | 0 |
| 3 | 1 | 1 |
| 4 | 2 | 1 |
| 5 | 3 | 2 |
| 6 | 2 | 1 |
| 7 | 1 | 2 |
| 8 | 2 | 0 |
| 9 | 1 | 1 |
| 10 | 2 | 1 |
| 11 | 1 | 1 |
| 12 | 3 | 1 |
| 13 | 1 | 1 |
| 14 | 3 | 1 |
| 15 | 2 | 2 |
| 16 | 3 | 1 |
| 17 | 2 | 2 |
| 18 | 2 | 2 |
| 19 | 3 | 3 |
| 20 | 2 | 1 |
| MEAN (Number and %) | 1.95 | 1.2 |
| Standard deviation | 0.759155 | 0.767772 |

A 2-sample t-test showed that spoken commands omitted more items of data than typed data [t (19) = 3.68, p= .002].

#### 4.2.3.4 Lexical commonality of spoken and typed words and sample-size effect

For each participant, the number of words which were only spoken, only typed or used in both modalities was calculated. The summed results are shown in Figure 4.12

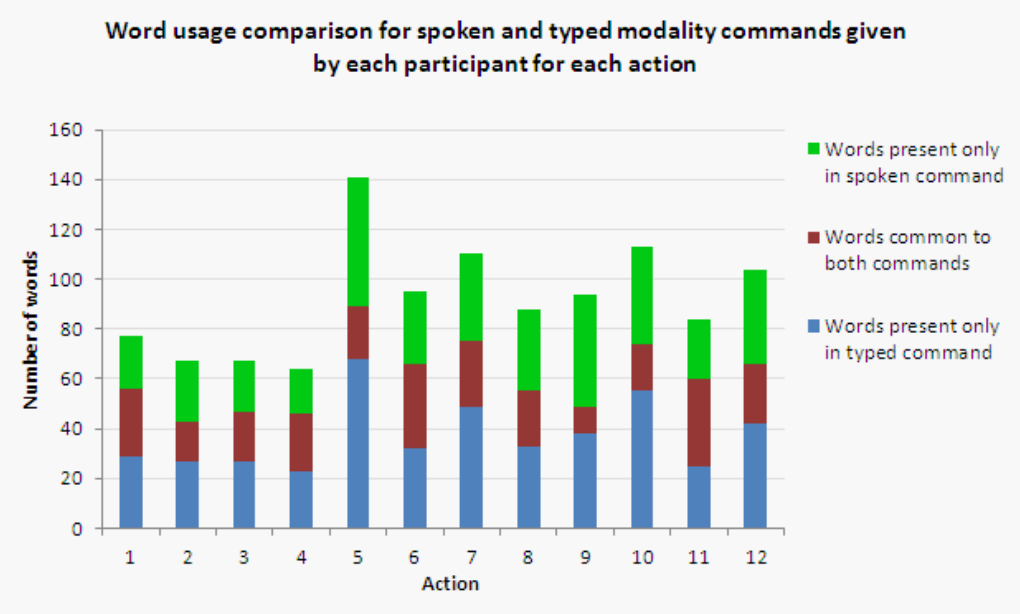



Figure 4.12. Count by action of common and modality-specific command words for each participant (study 3).

The words spoken by all participants were grouped for each action. The number of words (in the set for each action) which were only spoken, only typed or used in both modalities is shown in Figure 4.13.

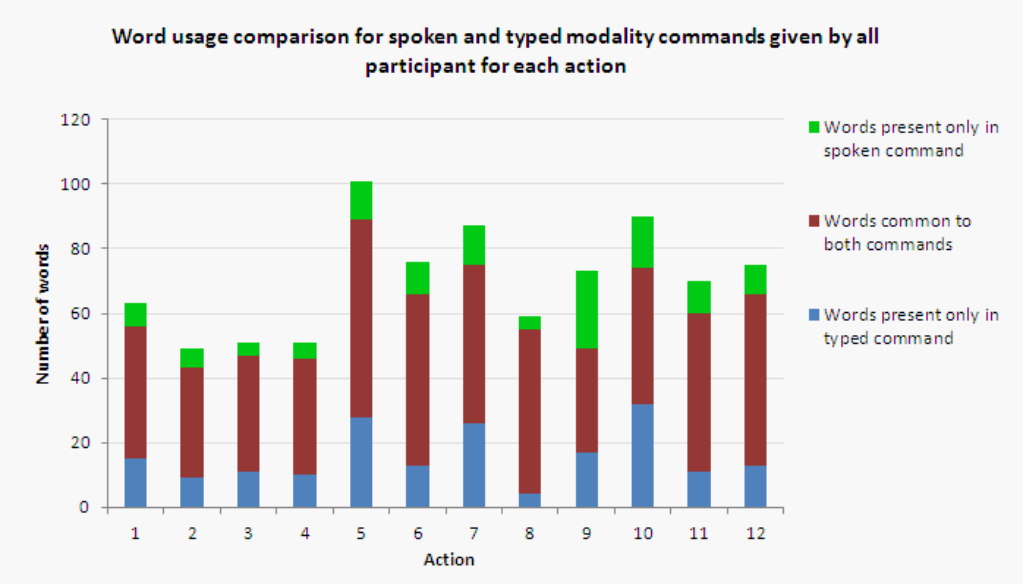



Figure 4.13. Count by action of common and modality-specific command words for all participants (study 3).

Comparing the words spoken by all participants with the words typed by all participants shows a significant increase in the proportion of common words. This shows that words used by a participant in one modality are more likely to be found in a set of commands provided by more than one participant in another modality. However, there remain words unique to the two modalities.

Commands may use some words which are common to two or more actions and some words which are unique to individual actions. As the number of participants' increases, the occurrence of a word (for a specific action) not yet spoken or typed by any preceding participant will become less common. The rate at which unique words are generated as sample size increases may be a contributory factor to understand the rate at which increasing sample-size can improve pattern-matching command processing.

Figure 4.14 shows the average number of new words generated by increasing the number of samples. This was generated by comparing all possible combinations of each stated number of samples for all actions.

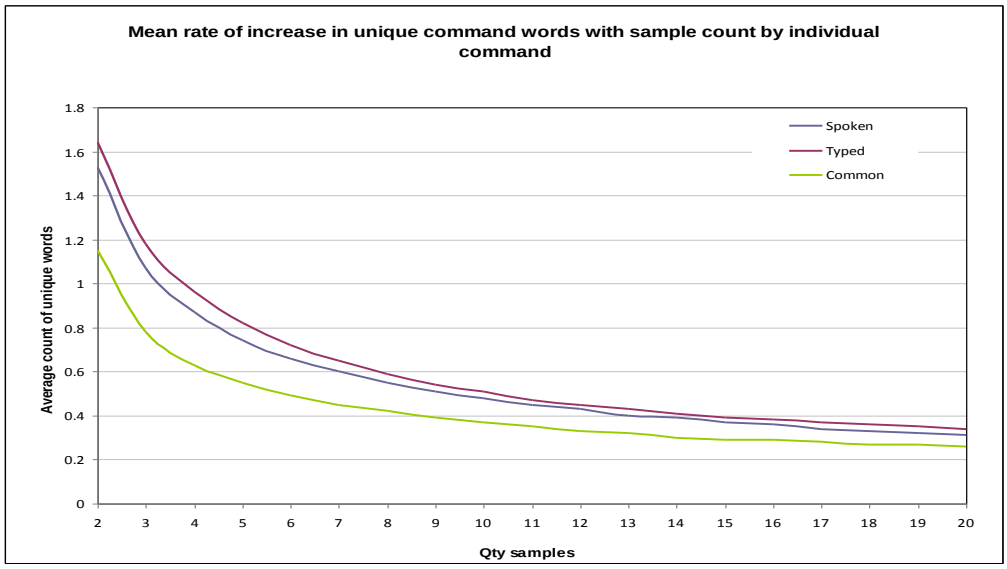



Figure 4.14. Correlation between mean count of unique commands words and quantity of command samples (study 3).

The spoken result shown in Figure 4.14 was calculated individually for each of the 12 actions, the result is shown in Figure 4.15.

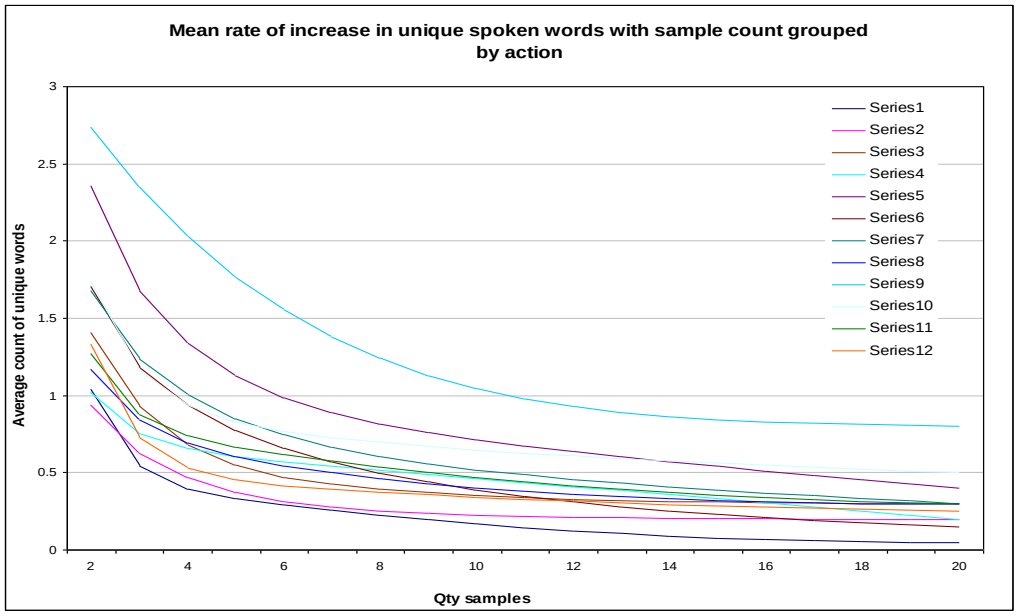



Figure 4.15. Correlation between mean count of unique commands words and quantity of command samples subdivided by actions (study 3).

One factor which may contribute to the variation in results for each action shown on Figure 4.15 is the number of words needed to fully define an action. Average word count for actions was compared to the average unique word increase on increasing sample-size from 19 to 20 (Figure 4.16). Correlation was found for uniquely spoken words [$F(1,10) = 55.59$, $p < .001$] with an $R^2$ of .848, uniquely typed words [$F(1,10) = 61.13$, $p < .001$] with an $R^2$ of .859 and

words common to both [$F(2,9) = 82.99$, $p < .001$)] with an $R^2$ of .949. Results are shown in Figures 4.17 – 4.19 respectively.

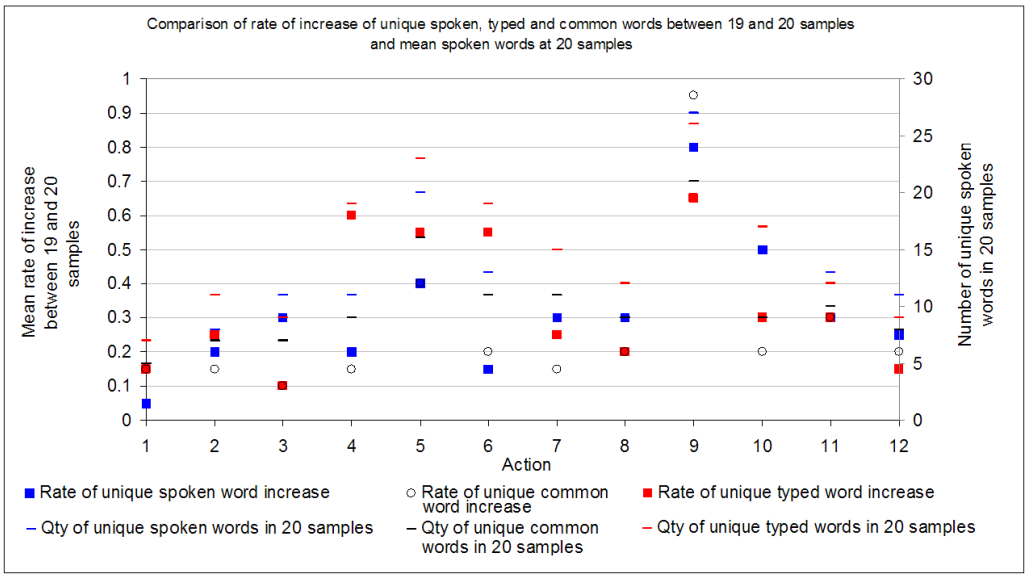



Figure 4.16. Correlation between the rate at which additional samples generate new spoken words and the quantity of words in a command for each action (study 3).

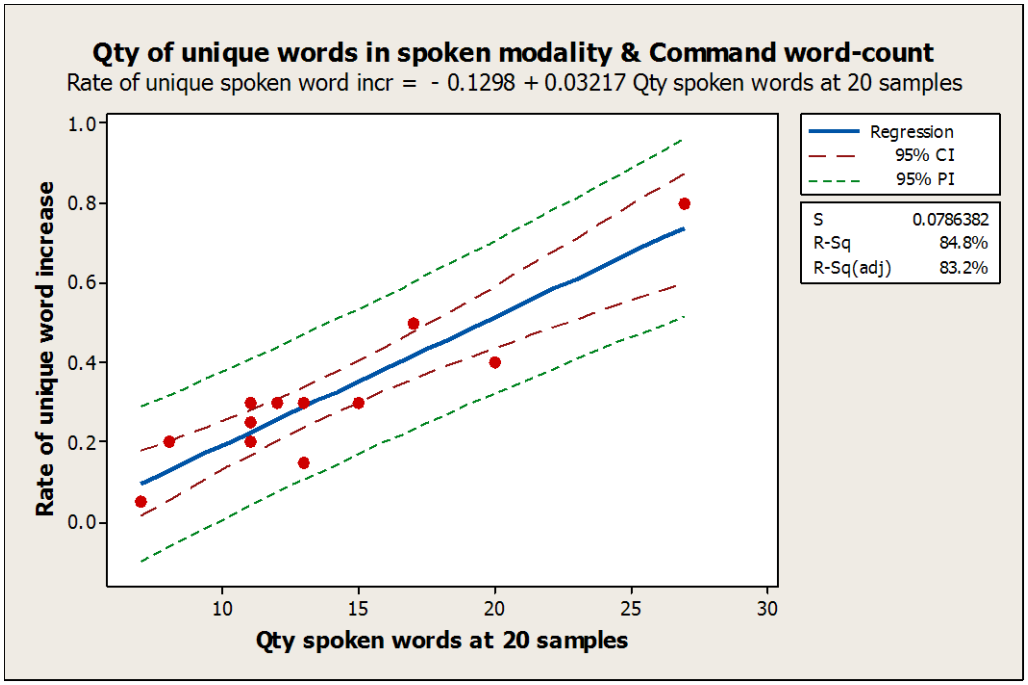



Figure 4.17. Correlation between the rate at which additional spoken samples generate new spoken words and the quantity of words in a spoken command (study 3).

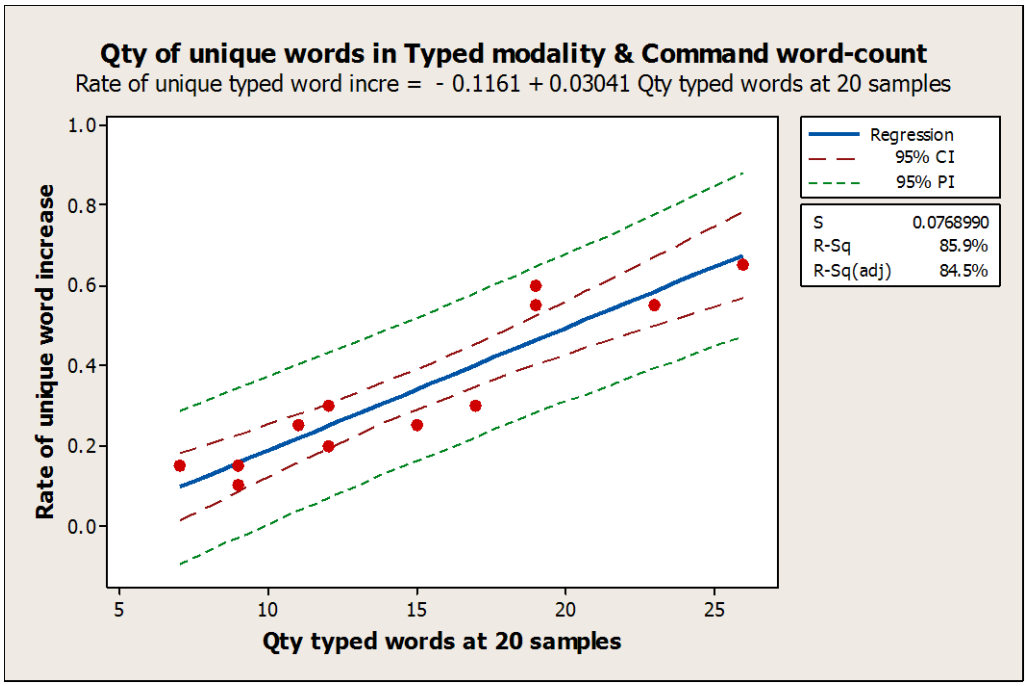



Figure 4.18. Correlation between the rate at which additional typed samples generate new spoken words and the quantity of words in a typed command (study 3).

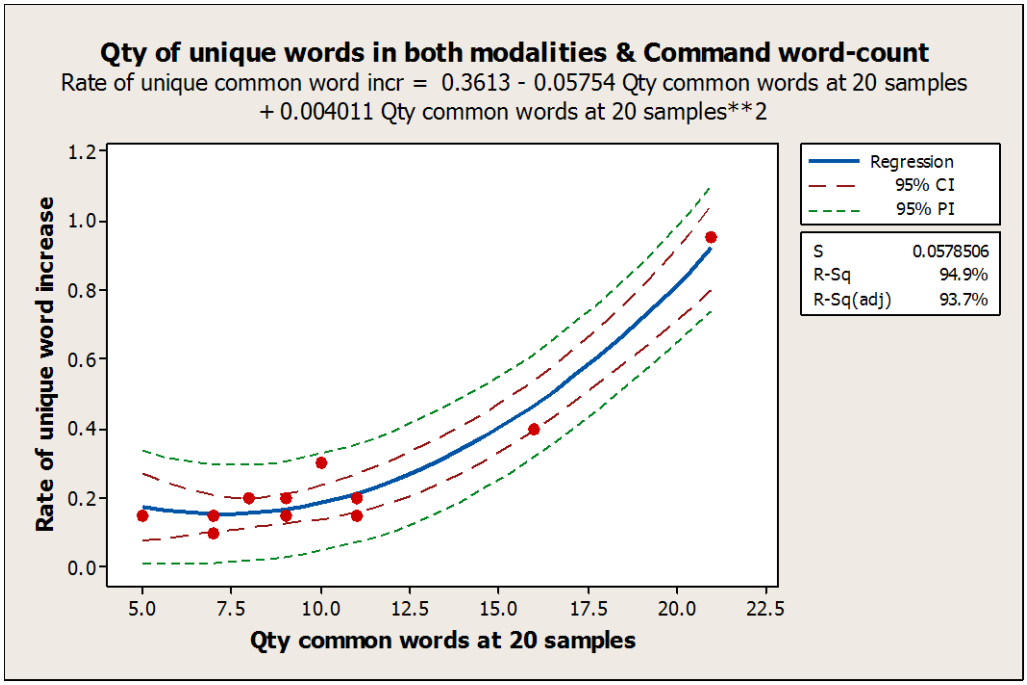



Figure 4.19. Correlation between the rate at which additional common samples generate new spoken words and the quantity of words common to typed and spoken commands (study 3).

## 4.2.4 Discussion

### 4.2.4.2 Modality and sample-size effects

The operation of a natural-language command interface may be affected by the degree to which users give commands which fully define a system and function and whether sufficient information is supplied to direct and quantify the action required. In this study, 26% of

spoken commands failed to clearly define the system and function from which an action was required, suggesting that in these cases a disambiguation action would be needed. All twenty of the commands given for one example action failed to supply all three items of information needed to fully define the action required. In all cases therefore, a clarification action would have been required before the command could be enacted. These results emphasise the need for effective disambiguation methods.

Time required to enact commands (when using a voice interface) was noted as important by participants in study 1. The results of this study suggest that a proportion of commands will require disambiguation which may increase time to enact a command. Users may, after a number of interactions, become accustomed to the level of precision with which they need to express command. The initial level of disambiguation required may therefore decrease over successive uses.

Typed modality commands were shown to have a higher word-count that spoken modality commands (Table 4.2). Results also show the degree of word commonality between typed and spoken commands. Figure 4.12 shows low word commonality between the first and second command given by the same study participant for the same action. Two options for this difference are that participants gave differing commands at differing times for reasons not related to modality; an alternative conclusion is that input modality affected command phrasing. In either case, it was found that participants vary their command phrasing and it cannot therefore be assumed that the command words a participant types can be used to identify by pattern-matching a command they then give using a spoken modality.

Figure 4.13 shows that a corpus of 20 typed commands includes the majority of command words spoken by participants. This suggests that increasing the number of samples in a corpus may increase the likelihood of pattern-matching a spoken command. Regression analyses show that as average unique word-count for an action increases, more samples are needed for the number of new unique words generated by a new sample fall to a common threshold. The average number of new words added to a corpus by each new sample decreased with each successive sample. Since additional command words may increase the chance of identifying patterns in a command, it may be necessary to identify when the probability of identifying new command phrasing drops below a threshold level.

One factor which might assist with identifying how many corpus samples are needed for an action is the average number of command words needed to express that action. Graphs 4.16 – 4.19 showed a correlation between average numbers of command words needed to express an action and the average number new command words in the twentieth sample.

Pattern matching may include sub-sentence patterns of words rather than individual words. In such cases, this may affect the probability of such patterns occurring with successive samples and therefore the relationship between quantity of samples and action-identification.

#### 4.2.4.2 Sample collection methods

One reason for introducing animated prompts was to encourage participants to focus on the intended system and function rather than the pictured conventional interface as used in study 2. This was successful as, whilst participant commands were in some cases ambiguous, the intended system and function was in all cases either clearly indicated or one of the possible command interpretations.

In study 2, command length averaged less than two words and one objective of study 3 was to increase this by using improved prompting to allow selection of more complex actions. In this study, commands averaged more than three words and the objective was therefore met.

The actions used in this study were selected from those which could be reasonably represented via animation to participants without conditioning their responses. For example, the display of words was avoided as these might then be used in a command or might prompt commands using a similar verbosity or style of speech. This effect was noted by Baber et. al. who stated that "use of mixed auditory and visual feedback improved user performance, such that out-task vocabulary was removed, " [7, p.56]. Whilst animations could enable a larger range of actions to be prompted compared to the static images used in study 2, it may not be possible to prompt all possible actions without including text in the prompt. For example, a textual prompt might be needed to clearly communicate the action of selecting a specific music play-list.

#### 4.2.4.4 Command processing method

The degree to which differences found between the spoken and typed modality samples may affect the confidence with which a natural language command processing system can select an intended action is not shown by this study. If degree of action-identification confidence is reduced, this would mean that response to the user for commands close to a decision threshold might be degraded, resulting in reduced habitability. For example, if a decision confidence threshold was placed between enacting a command or instead, first asking for confirmation (a degraded response), a command close to this level may fall below if a typed command corpus was used. However, a command which generated higher confidence might nevertheless remain within the better response-range effect.

To assess the effect of any degradation, the number of commands crossing response thresholds would need to be considered as a proportion of all commands (possibly sub-grouped by language, dialect or individual user). In order to fully investigate the effect of modality differences of this type, a command-processing method would be required.

### 4.2.5 Conclusions

This study sought to identify whether a set of crowdsourced typed modality commands could be used as a corpus for speech recognition-based natural language processing. A number of differences were identified which suggest that typed commands do not use identical wording to spoken commands, either for individual users or for combined quantities of samples from several users. It has also been shown that these differences reduce with increased sample quantity.

The degree to which the differences noted in the measures used in this study might affect habitability for a natural language pattern matching method will be specific to the method and can therefore only be determined with a specific tool to assess this. However, the fact that there are measurable differences between samples from the two modalities suggests that an assessment may be needed before typed modality data can be implemented as a corpus for any specific spoken-modality natural language command processing method. Chapter 5 describes the development of a simple pattern-matching command processing method which derives confidence levels when selecting actions based on spoken commands using a command-sample corpus.

# *Chapter 5: Identifying the action intended by a voice command: deriving a normalised confidence value*

## 5.1 Introduction

### Aim of Chapter 5

Chapter 5 addresses research question 3 (section 1.4); development of a method which is able to compare a command phrase to a set of possible actions and, by making use of a corpus of crowdsourced command samples, derive a normalised value for each action which correlates with the potential that the action is the one intended by the command phrase. The command identification method is required to be sufficiently sensitive that it can assess the effects of corpus modality in relation to sample gathering method, quantity of samples and interactions with incorrect/incomplete spoken word recognition (research goals 1, 2 and 4, section 1.4).

Specific aims were to:

1) Examine how effectively an empirical pattern-matching model was able to identify the actions intended by command phrases.
2) Evaluate methods by which the information contained within command samples might be used to identify an intended action. Sanders & McCormick [63, p.4] describe measurement criteria as:
    i) reliability (the results should be stable across repeated administrations)
    ii) validity (the technique should measure what it is really intended to measure)
    iii) sensitivity (the technique should be capable of measuring even small variations in what it is intended to measure)
    iv) freedom from contamination (the measure should not be influenced by variables that are extraneous to the construct being measured)
3) Explore the degree to which recognition-confidence differs between command phrases and therefore how sensitive this method may be to modality and other effects such as variation in pronunciation and incorrectly recognised words.
4) Normalise confidence levels (across all actions) that an intended action has been identified.
5) Optimise the use of the crowdsourced command samples to maximise sensitivity in identifying an intended action.

Since this thesis considers methods to allow a natural language voice command interface to operate in many different dialects and languages without incurring scaling costs, the method was required to use only data which was limited to one of the following criteria:

- Possible to gather using a crowdsourcing method from study participants (with internet access) from any geographic region.

- Available in the public domain or otherwise easily obtainable for all geographic regions
- Common to all geographic regions (and therefore needing only one data-gathering task to cover all regions)

### Selection of a pattern-matching methodology

The modality in which a command is given may affect the command phrase. Using a corpus of a different modality to that in which commands are to be stated may therefore reduce the confidence with which a natural language command identification method is able to isolate an intended action. Since a natural language command identification method was not available when the results from studies 2 and 3 (described in sections 4.1 and 4.2) were considered, it was not possible to directly evaluate the effect of differing command and corpus modality on confidence in identifying an intended action. The more simple measures used to assess the results of these studies indicated differences between modalities but not the effect of these differences. The method developed in chapter 5 was designed to consider the effect of differences in sample modality.

In the limiting case, a sufficiently large quantity of command samples might be collected such that the chance of any command precisely matching one of them would rises above the user-acceptable threshold for correct action identification. However, it seems likely that this would require an unfeasibly large number of samples and in addition, a phrase-matching method based on such a corpus would not be robust to voice recognition issues which cause words to be omitted or incorrectly identified. It was therefore necessary to consider methods which enabled two differing command phrases to be compared using an objective measure such that if the phrases intend a similar action (within the usage context), this would result in an increased probability of generating a higher value than in cases where the phrases intend different actions.

It was observed from studies 3 and 4 that whilst study participants from a common geographic location used a range of phrases for an action, these tended to be constructed from a common set of words or sentence fragments. One example is the activation of hazard lights where one study participant stated ‘emergency lights on’, another stated ‘hazard warning lights on’ and a third stated ‘turn on hazard lights’. The action-identification method for the method developed in this section was therefore based on modelling the expected number of matching words between a command and the corpus of command samples for each action and is therefore a token-based command identification method. In developing this method, different comparison methods were assessed and this is described in section 5.3.1.

### User interface considerations

Methods by which the user of a control method can be informed of the effect of a particular control will in some cases depend on the type of control modality. In the case of conventional

button controls, a symbol on the button is commonly provided as a guide. Multi-use controls may have screen displays indicating button effects. Screen based controls may have graphical indications or text labels which inform users of both the effect of a control and show the range of controls available.

One option for a voice-interface system to inform users of the available actions is to provide a screen display of valid commands. Whilst displaying valid commands may assist in identifying the specific words and phrases needed by a constrained-syntax system, this method requires drivers to look away from the road and may negate some of the potential benefits of voice interfaces. In the case of a fully habitable command processing method which allows users to give commands using any phrasing they wish, a method of locating a control or identifying specific phrasing is not necessarily required. In this case, the driving context which elicits the need for an action results in the driver deciding on a command phrase. The implicit prompt is an open question "what would you like the vehicle to do?"

## 5.2 Scope

A possible high-level process flow for voice-interface operation is shown in Figure 5.1.

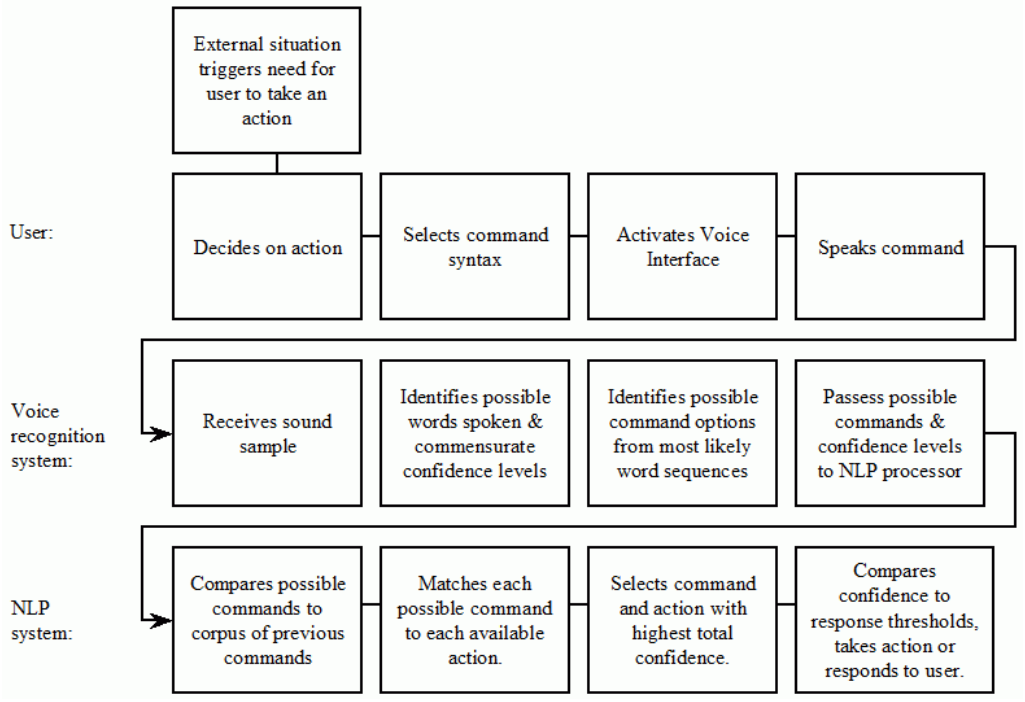



Figure 5.1. High-level process map for voice-interface operation.

The voice recognition system is not within the scope of this thesis and to consider the command processing and user-response methods in isolation, it was necessary to consider the outputs from a voice recognition system. Since spoken speech cannot be perfectly transcribed (in a limiting case, a extraneous noise might entirely mask a command), a spoken sound may match more than one potential word with each candidate word having a level of confidence

dependent on the degree to which the sound matched a language model. There may also be a confidence level that an individual word was spoken at all, the alternate hypotheses being an extraneous sound, a verbal tick or that the sound was actually part of a preceding or following word.

Following the process described by Pinto and Sitaram [72, unformatted], all of these potential options with corresponding confidence levels could be passed to a command processing method to identify candidate actions and confidence levels. The hypothesis in this case being that the intended command phrase is most likely to produce the highest overall command identification confidence level.

The process map in Figure 5.1 shows two areas where methods are needed:

1. Calculation of a confidence level that a command intends one of a set of actions.
2. Identification of the optimal response to the user based on a calculated confidence level.

Identification of optimal responses first requires a confidence level that each possible action is the one intended by a command. This section describes the development of a command processing method using the command samples obtained in studies 2 and 3 (sections 4.1 and 4.2). Chapter 8 describes study 6 which addresses methods of selecting systems responses.

The following scoping limits were applied when developing the method described in chapter 5:

1. Sources of information such as vehicle state, usage history, driving state and history and environmental state and history were not included in this method. The potential for this additional information to enhance command processing is considered in the discussion in section 5.8.5.
2. Spruit shows that syntax varies with geographical location [14, p.10]. One consequence of this is that before this method could be successfully employed for users with different patterns of speech, it may be necessary to identify the region from which the user and therefore the speech patterns originate and selection of a subset of samples which use similar command phrasing. The method developed in this section does not consider this requirement and is based on the assumption that the person giving the command and those who provided the corpus use a dialect which originates from a common geographical area. Identification of samples which are relevant to the originating geographical area and which therefore match user speech patterns is considered in section 5.8.5.
3. It was assumed that commands are given as a discrete input; no consideration is given to methods of isolating a command from other speech.
4. It was assumed that commands had been correctly transcribed by the voice interpretation system. Consequences of a word being incorrectly recognised are

considered in chapter 7 and methods to work with voice recognition methods which generate a range of word options tagged with confidence levels are described in section 5.8.3.

The following list shows factors and questions which were considered during the process of developing a simple pattern-matching command processing method:

- Pattern-matching may result in matches to the corpus for the intended action (signal) but also other, non-intended actions (noise).
    - What is the relationship between sample size and signal/noise ratio?
    - How will the signal/noise ratio scale with number of samples?
- How can a statistically valid confidence level be best derived from a word/sentence-fragment matching method?
- How can information which indicates degrees of effect (for example, the number of decibels to raise the music volume) be used to decide on the action required? I.e., should degree of effect be included in action differentiation or separated via pre-processing?
- Commands which have an increased number of words may consequently have an increased number of matches.
    - Is it necessary to normalise actual matches by potential matches?
    - Since not all words will be valuable in differentiating actions, do only the effective words need to be used as potential matches?

## 5.3 Development stages

### 5.3.1 Methods of pattern-matching and affecting factors

Methods available for comparing a command phrase to a set of sample commands included:

1. Word count. Many commands may have a similar word count so this does not indicate a likely candidate.
2. Word matching. Commands generally state a noun and a verb indicating the intent of the command and contain other content specific to the action. Word matching may match these words more commonly for the intended action.
3. Word root matching may be a good basis for comparison but removing the additional word content excludes data which might assist with action identification.
4. More complex matching sub-sentence patterns (e.g. word sequences) may give a higher confidence that the action is intended. However, matching only higher complexity patterns will reduce the number of potential matches and therefore the quantity of measurement samples.

Factors which may affect pattern matching include:

1. A politeness prefix was only noted in one of the commands given in study 2 where the prompt referred to a vending machine coin-return function and the participant gave

the command ‘Return change, please‘ (section 4.1). This is not therefore likely to be an important factor.

2. Plurality suffixes allow action differentiation in some circumstances. For example, the noun ‘wiper’ may indicate a rear system (where there is only one rear wiper blade) whereas the plural ‘wipers’ may indicate the front system.
3. Grammatical word-types may have different degrees of word-match; considering this factor may therefore improve differentiation.
4. Words indicating degree-of-action may allow action differentiation in some circumstances. For example, if the word ‘degrees’ was used, the command might refer to the cabin temperature but it is unlikely to refer to raising the sun-roof (unless the person giving the command was remarkably precise).
5. Differing actions could have the same sub-sentence matches but occurring with a different regularity. These probabilities could contribute to calculation of overall action confidence.

### 5.3.2 Method development

Development of this pattern-matching method progressed through three stages. Initially, a simple count-of-matches was performed between a command and the total set of command samples. Results were summed for each action as a measure of the likelihood of that action being the one intend by the command. A binary deconstruction method was used both for the command and all samples in the corpus with the intention of maximising the potential for pattern-matches. Table 5.1 demonstrates the binary deconstruction method for the command ‘Open fuel flap’.

Table 5.1. Binary sentence deconstruction for command ‘Open Fuel Flap’.

| Decimal value | Binary value | Sentence fragment |
|---|---|---|
| 1 | 001 | Flap |
| 2 | 010 | Fuel |
| 3 | 011 | Fuel Flap |
| 4 | 100 | Open |
| 5 | 101 | Open Flap |
| 6 | 110 | Open Fuel |
| 7 | 111 | Open Fuel Flap |

The directional method discarded potential reverse order commands e.g. ‘Flap Open’. This decision was made to avoid significant rearrangement of the command words as this might increase matches to non-intended commands and in addition, given sufficient samples, it might be expected that these would provide any valid reverse patterns.

Whilst this early stage was able to identify intended actions in many cases, the resulting sum was not normalised and the information provided by the degree of pattern matching to non-intended actions was not considered.

The second version of the method was similar to the first, however the results for the action with the highest number of matching patterns was divided by the second highest number of matching patterns by individual participant. This generated a signal/noise value which noted the degree to which the action with the greatest number of matches could be differentiated from the next most likely. This method resulted in 32% of samples showing no sub-sentence matching to any of the non-intended action samples, resulting in a zero denominator value and an undefined value for signal / noise ratio. The problem with an undefined value can be observed by comparing example ratios of 0.33 / 0 and 2.00 / 0. The differing degree of pattern matching for the intended action will result in differing confidence in selecting this instead of the alternative action. However, the undefined value does not allow this distinction.

The third method was similar to the second with the exception that instead of dividing the action with the highest number of matches by the action with the second highest number of matches by each individual corpus sample, the set of scores were summed for all participants before the calculation (i.e. sum of matches between command and all corpus samples for the action with the most pattern matches / sum of matches between command and all corpus samples for the action with the second highest number of pattern matches). This reduced occurrences of undefined values and it was possible to evaluate the method. High degrees of variation were noted between different actions and the result remained insufficiently normalised. It was determined that the number of matches which could be expected (given a specific action) needed to be determined and that parametric models of these could then be used to normalise results. This resulted in the fourth version of the method which is described in the remainder of chapter 5. During previous iterations of command processing method development, it was noted that the incidence of pattern matching of two or more words was uncommon. To simplify analysis, single word matches were used for the fourth stage of method development.

## 5.4 Data-set

The data-set from study 3 (section 4.2) comprised command samples (e.g. 'reset trip meter to zero') provided by twenty-one participants for twelve different actions for both spoken and typed modalities, totalling five hundred and four samples. The study 3 data had several improvements over that obtained from study 2 (section 4.1):

- Automotive-context actions were used
- An improved prompting method was employed for data collection which resulted in participants consistent addressing the intended action
- The selected set of actions elicited more verbose answers

## 5.5 Method for generating parametric models

The earlier development methods (section 5.3.2) were revised and improved to consider:

- Making use of pattern matches for all actions when deriving a confidence score for each action.
- Developing parametric models of expected responses against which individual commands can be assessed.
- Normalising of results to allow the potential for response thresholds to be investigated.

This method derives a set of parametric models of number of matching words for each action which could be expected if a selected action is the one intended by a command. Commands are compared to these statistical models and goodness of fit is used to calculate confidence level for each action.

The process for generating the statistical models is demonstrated through stages 1-5 using a simplified corpus comprising of three actions and three participants. For example, action 1 could be the sun-roof opening to an inclined state. A command given for this by participant B (sample 1B) could be 'please raise sunroof to open position'.

### Stage 1: Selecting the command

Sample 1A is removed from the corpus and is designated as the command. Table 5.2 shows the table state.

Table 5.2. Example corpus of three participant commands for three actions with first sample removed.

| | Action 1 | Action 2 | Action 3 |
|---|---|---|---|
| Participant A | | Sample 2A | Sample 3A |
| Participant B | Sample 1B | Sample 2B | Sample 3B |
| Participant C | Sample 1C | Sample 2C | Sample 3C |

## Stage 2: Count of words common to corpus samples and sample A1

Each of the samples in the table is compared with Sample 1A and the number of common words counted. For example, sample 1B had two words in common with sample 1A, sample 2B had no words in common with sample 1A. Figure 5.3 shows the total counts.

Table 5.3. Count of example command matches to example corpus samples.

| | Action 1 | Action 2 | Action 3 |
|---|---|---|---|
| Participant A | | 1 | 1 |
| Participant B | 2 | 0 | 0 |
| Participant C | 1 | 0 | 1 |

## Stage 3: Calculation of average word matches

The column of values for each action are averaged. The results are placed in the row from which the command was taken. For example, action 1 is calculated by (Sample 1B common words) + Sample 1C common words) / (2 samples compared) = (2+1)/2. Figure 5.4 shows the resulting table, this now comprises the average number of common words between sample 1A and each action.

Table 5.4. Average of pattern matches for participant A commands to example corpus samples.

| | Action 1 | Action 2 | Action 3 |
|---|---|---|---|
| Participant A | 1.5 | 0.33 | 0.67 |
| Participant B | | | |
| Participant C | | | |

## Stage 4: Repeat process for all participants.

Row 2 of the table is populated by repeating the stage 1-3 process, designating Sample 1B as the command. Row 3 is populated similarly designating Sample 1C as the command. Figure 5.5 shows the resulting table.

Table 5.5. Average of pattern matches for all participants' commands to example corpus samples.

| | Action 1 | Action 2 | Action 3 |
|---|---|---|---|
| Participant A | 1.5 | 0.33 | 0.67 |
| Participant B | 2.33 | 0.67 | 0.13 |
| Participant C | 1.13 | 0.5 | 0.5 |

## Stage 5: Generate expected-value models for each action

Mean and standard deviation values are calculated for each column as shown in Table 5.6. The resulting set of normal distribution models comprise a fingerprint of the levels of pattern matching which can to be expected for each action for a command which was intended to enact action 1. Figure 5.2a shows the set of normal models for Table 5.6 graphically.

Table 5.6. Mean and standard deviation models for all example corpus actions.

| | Action 1 | Action 2 | Action 3 |
|---|---|---|---|
| Participant A | 1.5 | 0.33 | 0.67 |
| Participant B | 2.33 | 0.67 | 0.25 |
| Participant C | 1.13 | 0.5 | 0.5 |
| Mean | 1.65 | 0.50 | 0.46 |
| StDev | 0.61 | 0.17 | 0.296985 |

Figure 5.17

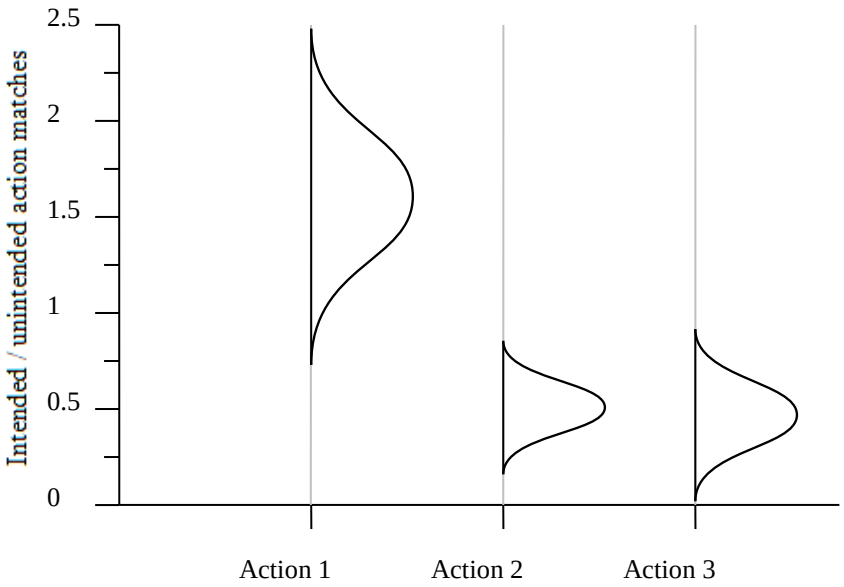



Figure 5.2a. Parametric models for commands which are intended to enact action 1; expected quantity of matches for each action.

### Stage 6: Repeat process stages 1-5 for all actions

Steps 1-5 are repeated for actions 2 and 3 to produce a set of statistical models for each action. Figure 5.2b shows a possible set of models for actions 1-3, a fourth action is added for clarity. The model for the intended action is highlighted in each case.

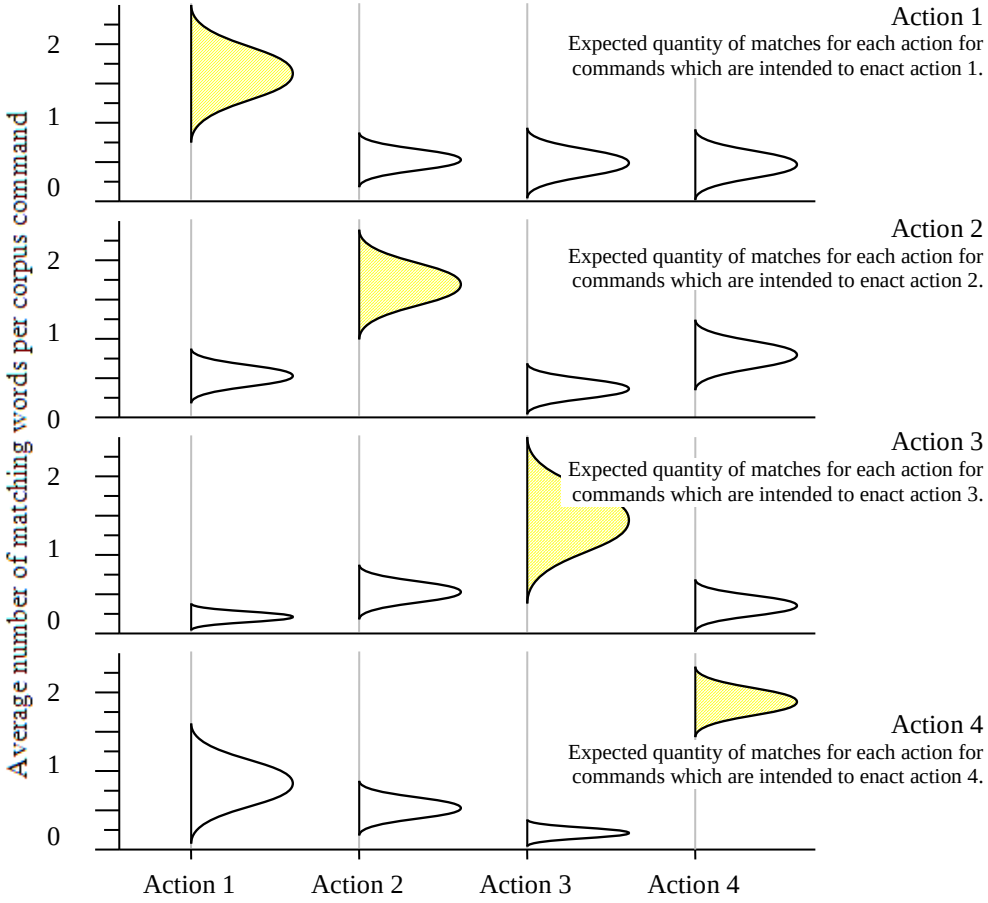



Figure 5.2b. Parametric models for commands which are intended to enact action 1-4; expected quantity of matches for each action.

If a command is intended to enact action 1, it is likely to have a high number of word-matches to corpus command samples similarly intending to enact action 1. It is likely to have a lower number of word-matches to corpus command samples for actions 2-4. Therefore, if a command is intended to enact action 1, the average number of pattern matches for each action is likely to fall within the models shown on the set of models labelled Action 1 on Figure 5.2b.

## 5.6 Method for command processing

The process for calculating a confidence value for each action is described by the flowchart shown in Figure 5.3. Steps 2-5 are demonstrated using a simplified corpus comprising of three actions and three participants. The numbers used for the example are synonymous with what might be expected for a command which is intended to enact action 1.

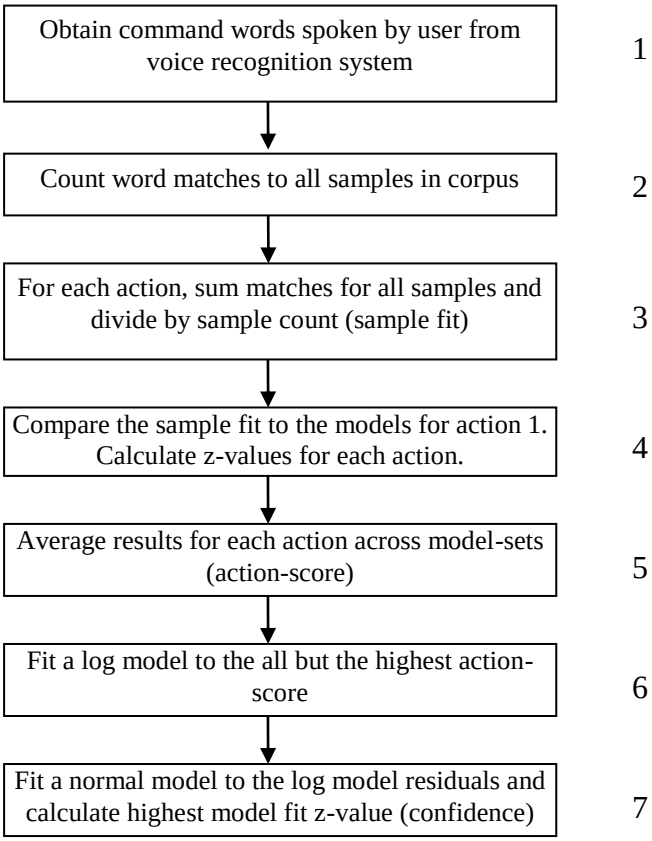



Figure 5.3. Flow diagram for calculating confidence in identifying intended action.

### Stage 1: Voice recognition

A system user speaks a command; the voice recognition system passes the words it identifies to the natural language processing method.

### Stage 2: Count of words common to corpus samples and command

The command is compares to each sample in the corpus. The number of matching words is counted. Table 5.7 shows an example for three actions and three samples.

Table 5.7. Count of user command matches to example corpus samples.

| | Action 1 | Action 2 | Action 3 |
|---|---|---|---|
| Participant A | 4 | 1 | 1 |
| Participant B | 1 | 0 | 0 |
| Participant C | 1 | 1 | 0 |

## Stage 3: Calculation of average word matches

The counts for each action are averaged in a similar manner to stage 3, section 5.5. Table 5.8 shows the result.

Table 5.8. Average of pattern matches for user command to example corpus samples.

| | Action 1 | Action 2 | Action 3 |
|---|---|---|---|
| Action 1 result | 2 | 0.67 | 0.33 |

## Stage 4: Comparison of average word matches to empirical models

Values in Table 5.20 are compared to the models established in section 5.5 and z-values are calculated for each action. The method is shown graphically in Figure 5.4 using the models shown in Figure 5.1. Where there are a limited number of degrees of freedom in the data-set, sigma values are increased according to the Student t-distribution. Average z-value is then calculated.

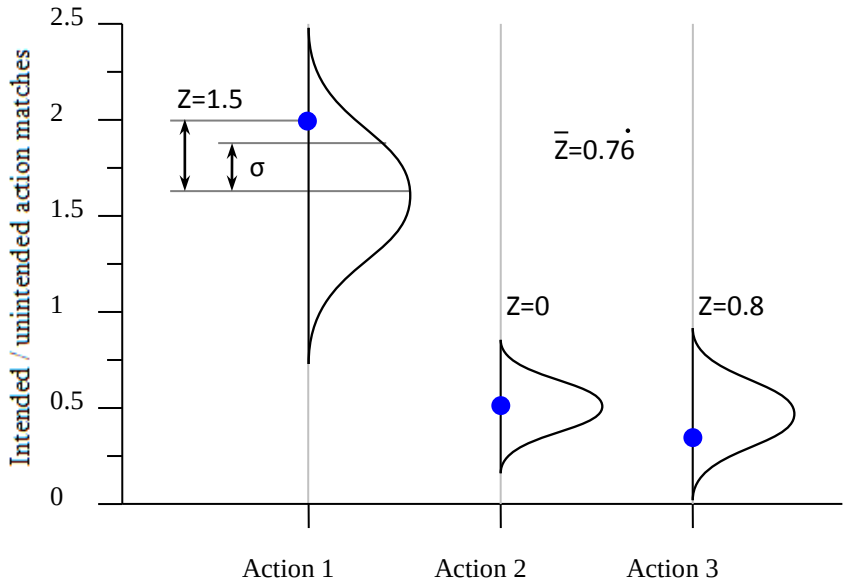



Figure 5.4. Z-values for pattern-matches to action 1 models for a command intending action 1.

Repeating the above operation when incorrectly assuming that the action 2 is the intended action results in a response shown as an example in Figure 5.5 and for actions 1-4 in Figure 5.6 (action 4 is added for clarity).

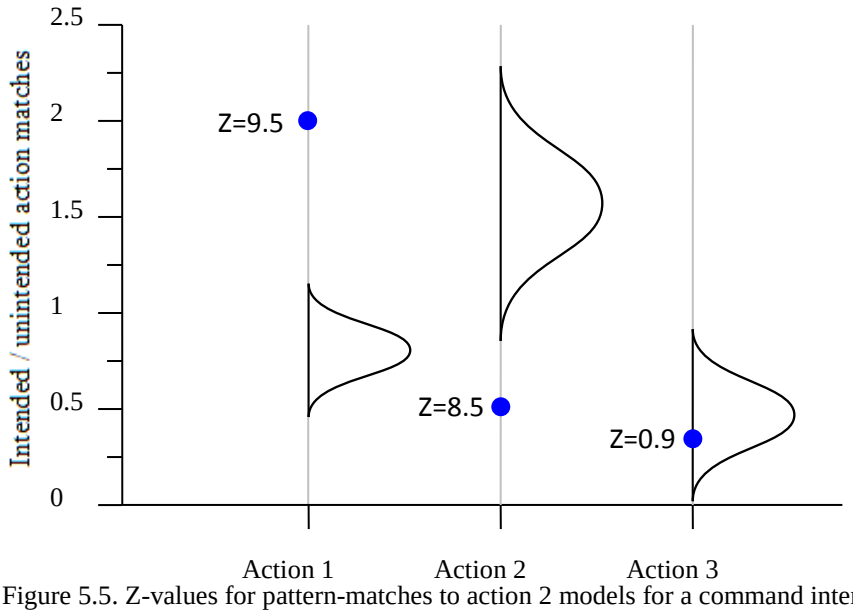



Figure 5.5. Z-values for pattern-matches to action 2 models for a command intending action 1.

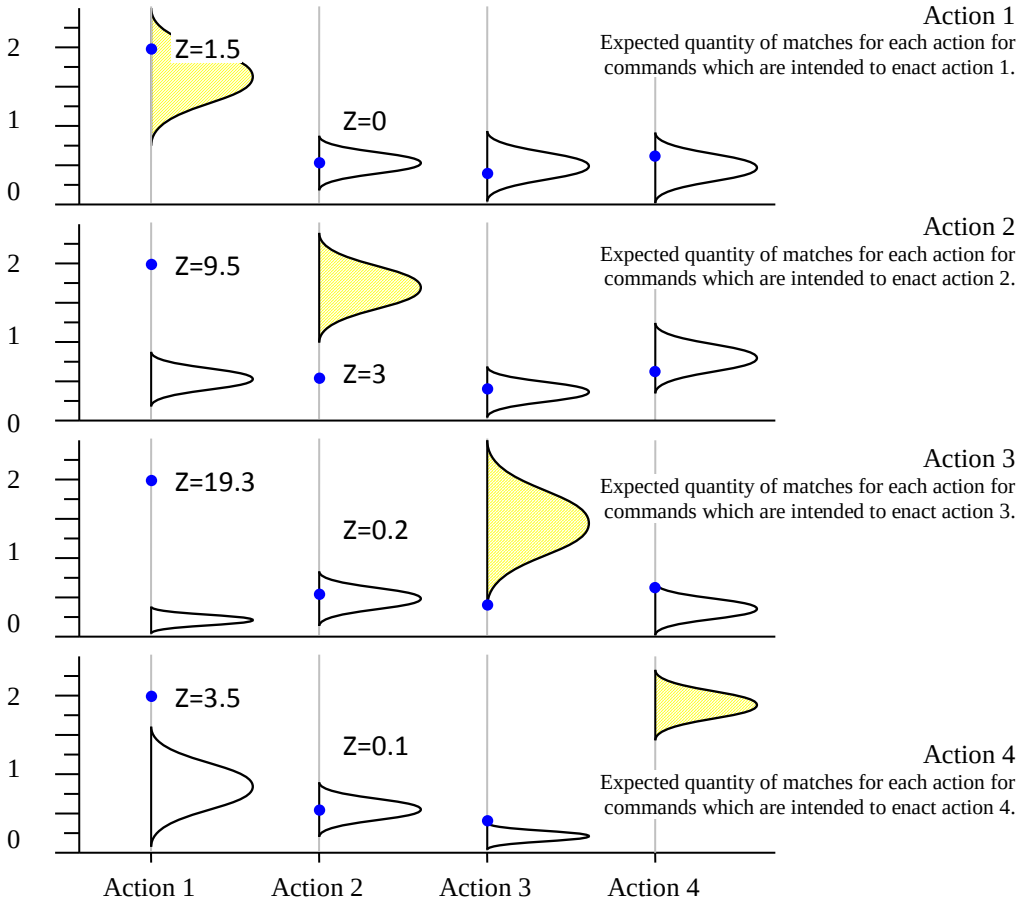



Figure 5.6. Z-values for pattern-matches to models for actions 1-4 for a command intending action 1.

## Stage 5: Cross model averaging

In this stage, the results for each action are averaged across the models. This can be seen graphically on Figure 5.6 where taken vertically (across models), it can be seen that the z-values for action 1 are low in the (one) case where action one is expected (z=1.5). For each of the remaining sets of models which in all cases expect another action to be intended by the command, the expected average word match is lower than the value obtained for the command (z = 9.5, 19.3 and 3.5). Averaging all the models for action 1 (using the results from Figure 5.7) shows an average z-value of 1.5 + 9.5 + 19.3 + 3.5 / 4 = 8.45. In the case of actions 2-4, the models all fit except one (where the models expect a high pattern-match which is not the case). For example, the result for action 2 (Figure 5.16) is 0 + 3 + 0.2 + 0.1 = 0.825.

An example using the data obtained during study 3 (section 4.2) is shown in Figure 5.7. Action 5 was intended by the command given in this case and it can be seen that averaging these results will show a high mean value for the intended action compared to the other actions.

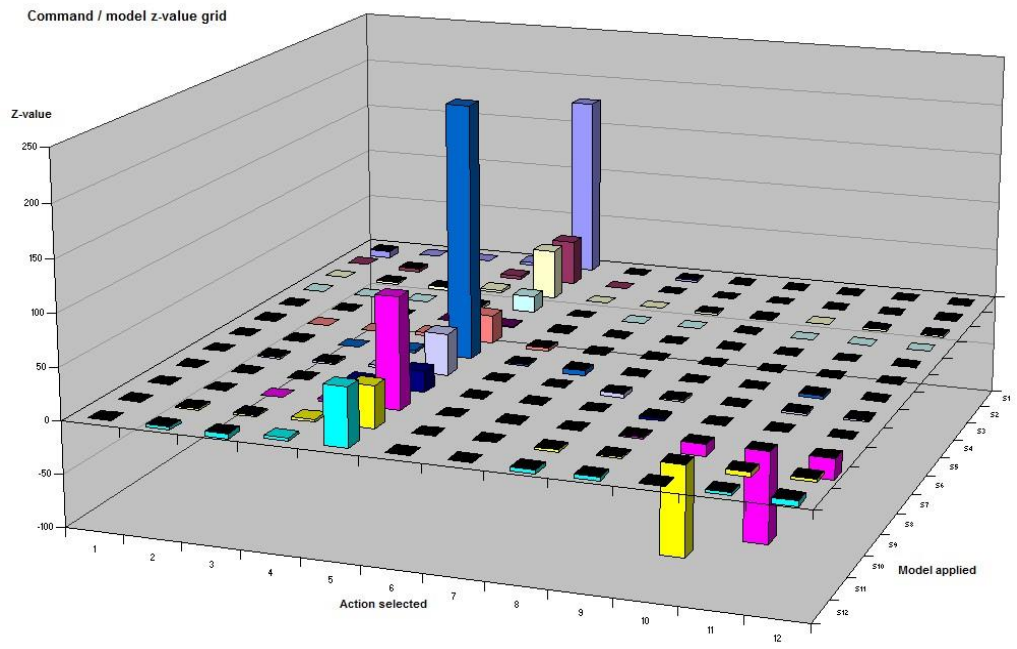



Figure 5.7. Z-values for pattern-matches for all models for a command intending action 5.

## Stage 6: Compensation for random signals

The process described in section 5.6 derived normalised values for the likelihood of each action being the one intended by a command. Since commands may vary in the degree to which they match the non-intended models, there may be random variation in non-intended pattern-matching level which could be conflated with signal. To evaluate this potential, each command in the corpus obtained from study 3 was assessed using the method described in section 5.5 and 5.6. The results for non-intended actions were ranked by mean z-value and shown as a set in Figure 5.8.

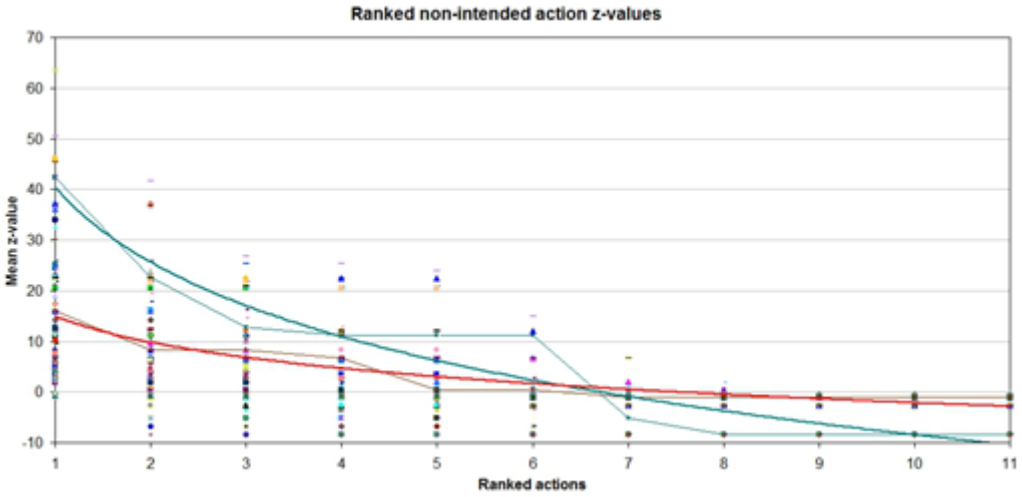



Figure 5.8. Z-values for all non-intended action for all commands.

A logarithmic model visually fitted the sets of results shown in Figure 5.8. Two example data-sets are highlighted by a line interconnecting the data-set and a thicker line showing the logarithmic fit in each case.

## Stage 7

Confidence level that the most probable action was the one intended by the command was calculated by firstly generating a normal model of the residuals (vs. a best-fit logarithmic model) for the 11 less probable actions and then by calculating the z-value for the (deleted) residual for the most probable action. This is shown graphically in Figure 5.9.

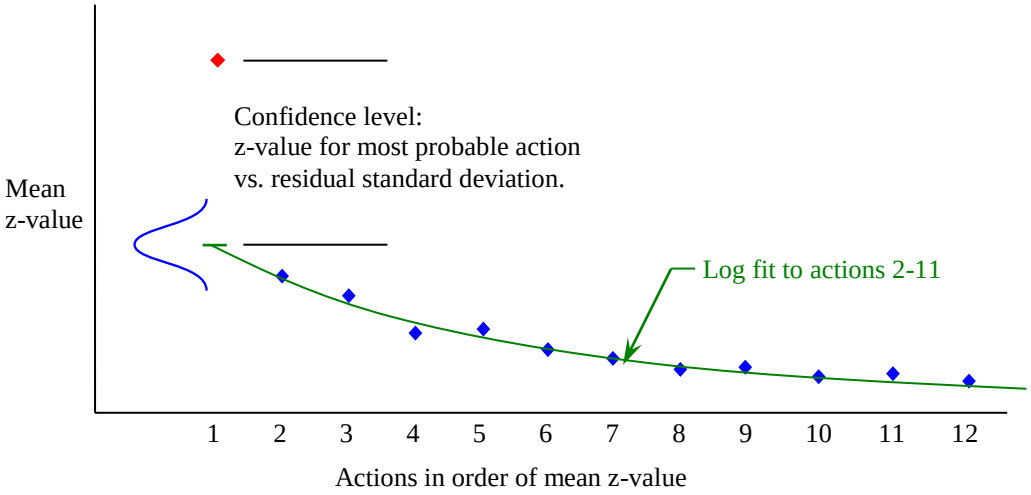



Figure 5.9. Calculation of normalised confidence level using logarithmic error-level.

To evaluate the potential for the most likely actions to be ambiguous, the same method was repeated using a model fitted to the 10 less probable actions and comparing the results for the most probable two actions. Figure 5.10 shows this process for a data-set where actions ranked at 1 and 2 are ambiguous.

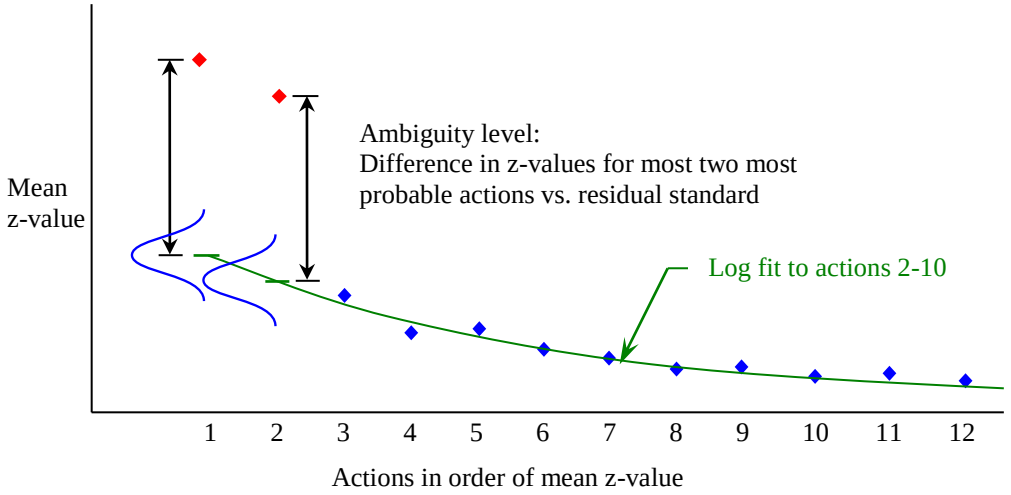
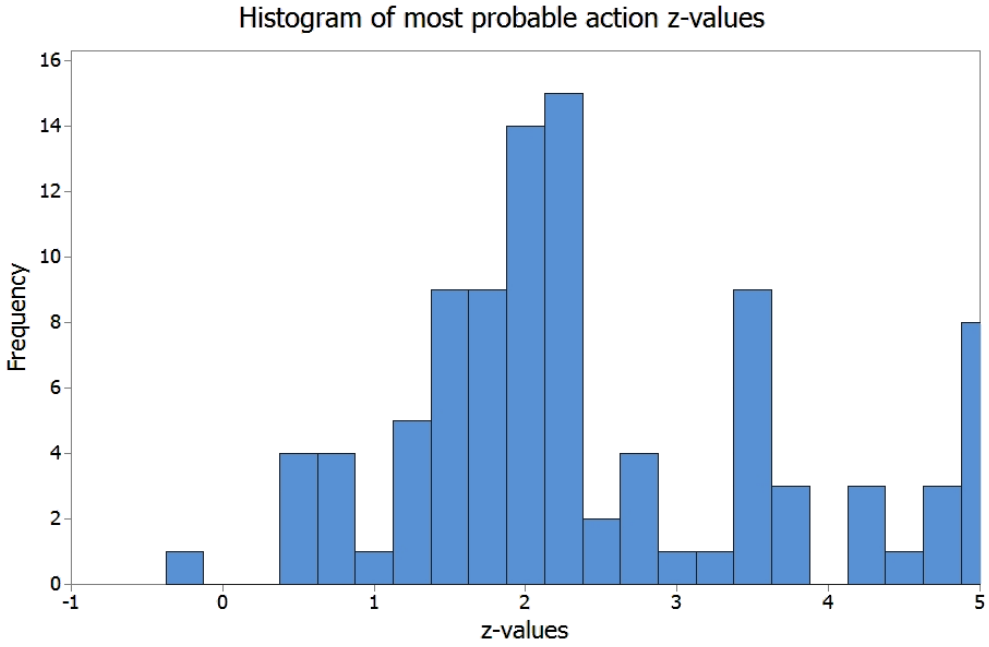



Figure 5.10. Calculation of normalised confidence level for two most probable actions using logarithmic error-level.

## 5.7 Results

Z-values showing confidence level relates to the probability that the most likely action is not a part of a model of less likely actions. Figure 5.11 examines the results of applying the method developed in this section to the 240 commands obtained in study 3.




Figures 5.11. Normalised confidence values for 240 commands, scale limited to 5.

Of the 240 samples tested, a number fell into the range where immediate action identification is unlikely. Figure 5.12 shows a comparison of the two most probable actions to identify potential ambiguity. It should be noted that the x-axis is constrained to a maximum of 20 to enable the low-level detail to be seen.

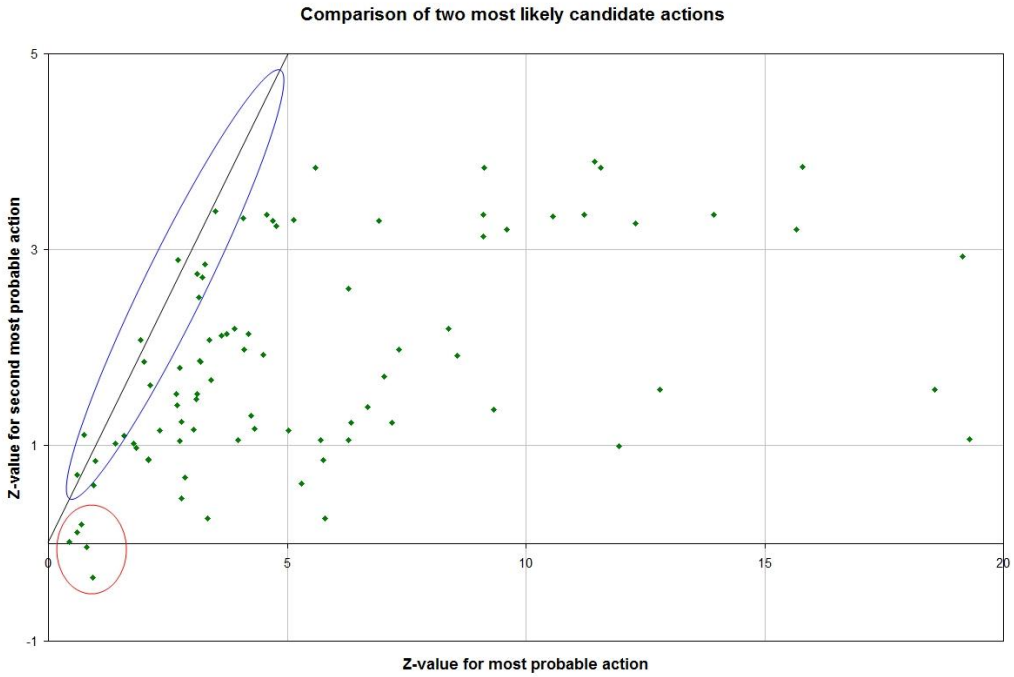



Figure 5.12. Comparison of normalised confidence values for two most probable actions for 240 commands, scale limited to 20.

The area highlighted in the upper ellipse (blue) in Figure 5.12 shows where the first and second most probable actions have similar z-values, indicating ambiguity. The lower ellipse (red) area shows values which are undefined; whilst they are not ambiguous, the low results may not clearly identify an action.

# 5.8 Discussion

## 5.8.1 Quality of method

The command processing method described in section 5 comprises an empirical statistical model which establishes a set of statistical patterns within which data points can be expected to fall if a command intends each of the available actions. It incorporates the understanding that some degree of pattern-matching can be expected between a command and the actions which it does not intend to enact and that these signals are able to contribute to the identification of the intended command. Comparing such results for each action allowed calculation of confidence that the most likely action was from a different population to the other, less likely actions. In each case, use of single data-points was avoided and the full set of information was used to calculate a normalised confidence metric.

Figures 5.11 - 5.12 showed that a high percentage of commands were identifiable. However, despite this improved use of response modelling and relatively data-rich calculation, a small number of commands within the corpus obtained during the study 3 (section 4.2) were not clearly identified, showing on Figure 5.12 as either undefined or ambiguous.

One possible reason for the small number of low z-values is the responses of the study participants. One participant was quite observational rather than command-oriented in responses to prompts. One example was the animation of the instrument binnacle brightening. A command such as 'increase dashboard illumination' might have been expected, however the participant in this case simply stated 'dial lights up'. This resulted in one of the points highlighted in the red oval in Figure 5.12. This suggests that it may be difficult to obtain useful samples from people who respond to prompts in this way. Whilst users giving unusual responses could result in an ambiguous or non-recognised command, allowing such responses to be included in the corpus used for command identification could adversely impact all users. Use of the entire corpus is intended to mitigate this potential as outlying results will average out.

Different actions generate models which vary considerably in the degree of expected pattern matching. They also differ in the degree to which pattern-matching can be expected from other, non-intended actions. Where two actions are likely to use many common words, there may be an inherent difficulty in isolating the intended action. This can be observed by considering the actual models obtained using the samples obtained in study 3. The mean value model is shown in Figure 5.13 and the standard deviation model is shown in Figure 5.14.

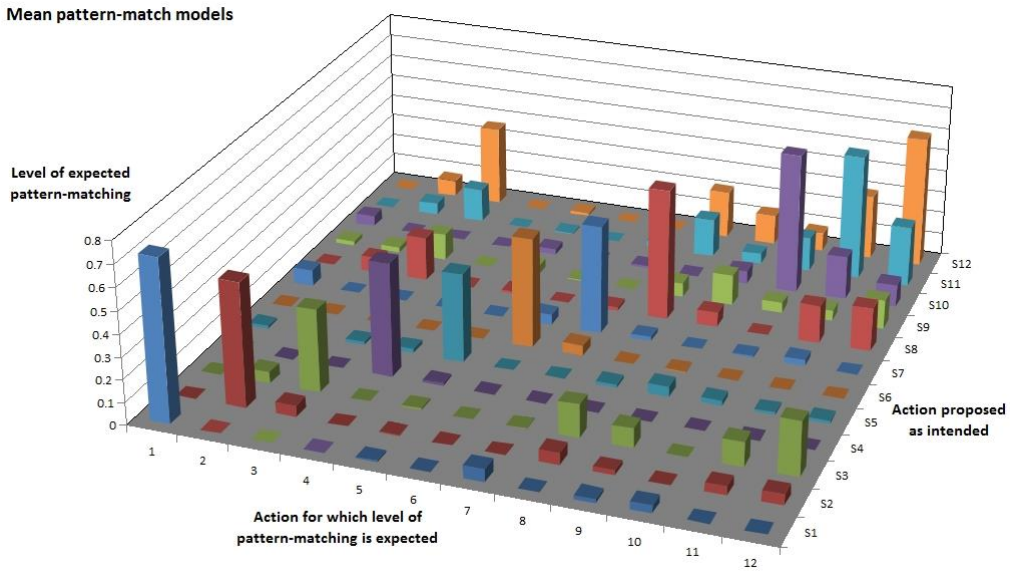



Figure 5.13. Mean expected value models for actions based on study 3 samples.

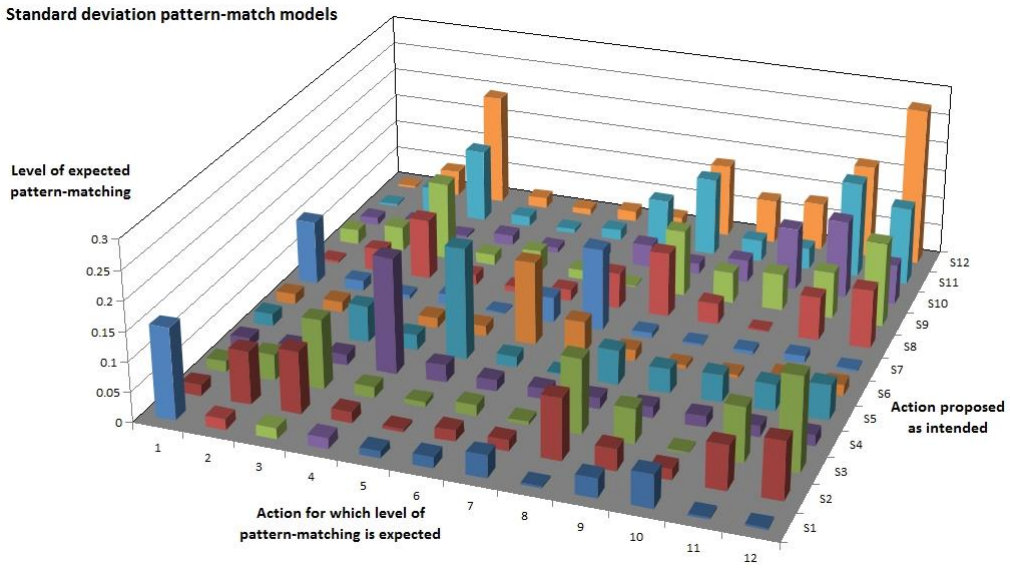



Figure 5.14. Standard deviation models for actions based on study 3 samples.

Figure 5.13 shows that if action 1 is that intended by a command (selected via y axis S1), the expected level of pattern matching for action 1 (shown on the x/z axis) is considerably higher than the other actions. The similar information for standard deviation (Figure 5.14) shows that variation for non-intended actions is relatively small and this suggests that action 1 will be easily identified.

By contrast, where action 12 is intended by a command, Figure 5.13 shows that both action 12 and action 3 expect a high score and the reverse is the case for action 3. Figure 5.14 shows considerable variation in the scores for these values and this suggests that a command

intending action 12 may have similar word sequences to and consequently, be inherently difficult to isolate from action 3.

The completed command processing method matched single words, a simplification of the binary sub-sentence combinations used in earlier development forms of the method. It would be possible to reintroduce this additional complexity if it were found to be useful in identifying actions for commands which could not otherwise be identified.

### 5.8.2 General discussion

In this thesis, it is suggested that a command processing system could be based on a corpus of crowdsourced example commands for specifically available actions, given by users within the context in which such a system would be used. This would mean that the system could only identify commands intending the fixed set of actions and that it would only be able to identify commands given by people speaking in a similar way to the providers of the corpus of example commands. Whilst this would seem to be limiting, the purpose of proposing the latter approach is to eliminate scalability problems by allowing a crowdsourcing approach to obtain the required data. To ensure this, the constraints placed on the corpus for method developed in chapter 5 were:

- Information which is commonly applicable to all languages and dialects and can be pre-set during method commissioning.
- Information which can be crowdsourced from multiple study participants remotely (i.e. via the internet, with no researcher present) and in parallel (all participants can respond simultaneously).

In chapter 5, the data from study 4 (section 4.2) was used; this data-set met the stated constraints.

To enable the results of these studies to be valid in the context of a potential future application, it was necessary that the resulting method was capable to a level of performance which could be practically applicable. Since there was no direct measure for this level of performance, the methods was instead developed to make full use of the available information and ensure that no obvious opportunities to improve the sensitivity presented themselves. Study 6 (chapter 8) describes one use of the method developed in chapter 5; to allow a crowdsourcing-based investigation into user-preferred thresholds for system responses.

At the start of method development, the level of complexity of the final method was not expected. A simple count of the number of matching sub sentence patterns seemed to provide a reasonable detection of the intended action. However, the need for a statistically meaningful and normalised confidence level on which a response prediction could be predicated considerably increased the need to account for differing degrees of pattern-matching, command

variation and ambiguous cross-matching of non-intended actions. The final command processing method met these requirements.

In section 5.1, a set of objectives for this command processing method were stated. Objectives 1 & 3-5 were clearly met by the method developed in this section. The second objective referred to the 'optimal measurement criteria' described by Sanders & McCormick [63, p.4]. These are listed and responses described as follows:

**Reliability** (the results should be stable across repeated administrations). Rather than note that the method would give the same answer for the same command, reliability in this case could be extended to consider a 'repeated administration' as a set of commands from participants of similar dialectic backgrounds intending a specific action. Such sets were shown to be differentiable from similar sets of commands for other actions and were therefore considered to be reliable.

**Validity** (the technique should measure what it is really intended to measure). The method has been shown to detect intended actions from a variety of command phrases, generating a normalised confidence value.

**Sensitivity** (the technique should be capable of measuring even small variations in what it is intended to measure). Confidence was affected in all cases by any change in command phrasing.

**Freedom from contamination** (the measure should not be influenced by variables that are extraneous to the construct being measured). The variables included in the method were commands given and the example commands of which the corpus comprised. The candidate for contamination was the proposed use of a typed modality corpus. This is considered in chapter 6 and is not a factor in the quality of the command processing method described in chapter 5.

### 5.8.3 Options for future consideration

It would have been possible to include additional information in the method without fully setting-aside the constraints of being either unrelated to geographic location or obtainable via crowd-sourcing. One example was inclusion of grammatical language types to allow shallow-parsing and sub-grouping of the data and therefore development of specific models for each type (and for interactions). This was not done in this case as, whilst the data could be pre-set, it was nonetheless different for each language and the task would scale with geographic area. It is however an option for future improvement.

Modelling of response distributions was performed using a simple mean and standard deviation and was based on an assumption of normality. With a larger data-set, the validity of this assumption could be better tested and if needed, the use of a better fitting model may be an option for future improvement. It may also be possible to compare the pattern-matching table for the corpus to the pattern-matching table for a command using non-parametric methods such as chi-squared.

The proposed method of obtaining a number of high-confidence word alternatives from the voice recognition method and putting all possible combinations to the command processing method could be subjected to study to ascertain the effectiveness of the method and identify any consequential effects.

### 5.8.4 Remaining challenges

The actions used for development of this method required data in some cases. One example was the action of setting the ventilation fan to its highest speed and the in-cabin temperature to 22 degrees. One participant gave the command 'increase fan speed to max and increase temperature to 22 degrees' correctly identifying 'maximum' speed and '22' degrees as the required data whereas another participant stated 'increase fan temperature to 22', thus omitting the absolute fan speed date. Such commands were processed as stated for this development and no assessment was made of how to manage a range of alternative data forms, for example to set the fan to full, maximum or high. One option would be to use the method as developed with a data-free corpus to identify the command and then once the action was identified, to use the difference between the command and the corpus for that action to isolate the data. Under some circumstances however, use of units by the user may help in identifying the intended action. For example, a user might wish to 'raise the temperature by five degrees' but is unlikely to ask to 'raise the sunroof by five degrees'. If the subject of the command were obscured, the units may be enough to identify the intent in this case.

A method for identifying a disambiguation prompt is required. One possible option is to compare a command to the corpus commands for the two candidate actions and assess how adding each word from the corpus commands in turn would affect identification. The two words which most definitively identify the intended action could be presented to the user to select from. For example, should the user say 'turn the hifi on', the system may not be clear which entertainment function is required. Comparing the command to the corpus samples for the two types of hifi system may identify that adding 'radio' to the command strongly identifies that system whereas adding 'CD player' to the command similarly enacts the other system. The user could then be asked 'would you like radio or CD player?' An alternative method would be to pre-program a table of commands for each possible ambiguous pair. A third possible method might be a hybrid system which pre-programs responses where the first method proposed generates unhelpful automatic responses.

The method developed in chapter 5 was designed to identify the action intended by a user who is speaking a command while in an operational context (in this case, driving a car), is expecting a good response and is actively attempting to enact an action. It is possible that results may be adversely affected if a user is in a different context or had a different expectation or intent. For example, a user may initially test the system to see how well it works, wording a command with limited expectation. In this case, limited expectation may affect command phrasing and result in poor response and result in long-term performance degradation. One option to address this might be to collect a sample-set under this context and provide this as a parallel pattern-matching corpus for new users.

During the development of this method, the samples used were obtained from study participants who worked in a common location and could therefore be expected to have some similarity in command phrasing. In a live usage scenario, the samples of which the corpus comprises would need to be selected from a much wider set with the selection predicated on the potential origins of users. Identification of the geographic origin of users would be required and one option to do this would be to ask. Samples could then be selected by geographic proximity (assuming that each sample is geographically tagged). One option for further improving the sample selection would be to process additional samples when the system is not otherwise used to hunt for an ideal sample-set. This may be of particular benefit where users have complex linguistic backgrounds and one sample-set does not optimally replicate their choice of command phrasing. It may also be possible to use voice recognition techniques to identify a change of user or to distinguish between several regular users and thereby select the best sample-set.

### 5.8.5 Usage considerations

Voice recognition systems can produce a number of candidate words for each user utterance, each with a confidence or probability value. Whilst grammatical rules can be used to assist with selecting and scoring these options, it may be necessary to identify which of two or more similarly scoring words is intended and a command processing system may need this capability. In the case of the method developed in chapter 5, this can be simply achieved as phrases with all possible combinations of candidate words can be presented to the method and the phrase with the highest degree of action identification can be assumed to be the one intended.

Probability of correct action identification could be enhanced by taking additional context into consideration at the point at which a command is given. Examples include vehicle state, usage history, driving state and history or environmental state and history. These were not included in this method development as the requirement was optimisation based on the stated constrained set of information. It would however be possible to include additional modelling

based on observed context links during usage using a closed-loop algorithm. Whilst this may further increase probability of successful command identification and therefore habitability, the method developed without this benefit must nevertheless meet the minimum acceptable user performance to ensure usage persists.

## 5.9 Conclusion

The command processing method developed in chapter 5 completes research question 3 (section 1.4), to develop a method of identifying an action intended by a command (and confidence in that identification) based only on crowdsourced command samples.

The purpose for developing the method in chapter 5 was to allow investigation of the effect of differing command and corpus modality on action identification and also to allow the human-machine interface to be studied with the potential to consider response thresholds based on confidence in action identification.

It has been demonstrated that a simple sentence-fragment pattern-matching method is able to identify actions from command phrases with a reasonably high level of syntactic habitability across several participants and actions. Results show method sensitivity in differentiating confidence levels between actions and participants and both validity and freedom from contamination where, whilst noting the difference between actions and participants, the method was nonetheless able to identify the intended command.

The method is therefore sufficiently sensitive for use in measuring the effect on action-identification confidence of modality and other factors such as variation in pronunciation and incorrectly recognised words.

In chapter 6, the developed command processing method will be used to assess the interface-level consequences of using a typed modality for the command processing method corpus. This will therefore consider the effects of modality differences identified in studies 3 and 4.

# *Chapter 6: Confidence level and probability of distinguishing between actions (study 4)*

## 6.1 Introduction

Aim of Chapter 6

Chapter 6 addresses research question 2 & 4 (section 1.4), to investigate the effect of modality on choice of command phrase and the effect of using crowdsourced, typed samples to interpret spoken commands.

Overview

This thesis examines the potential to use a crowdsourcing methodology to gather automotive voice interface command samples and in conjunction with a token-based natural language command processor, identify which of a set of possible actions is intended by a spoken command. Following an initial attempt to obtain spoken command samples, it was determined that this presented considerable difficulty (as a spoken modality is not a common home computer input modality). A typed modality was therefore selected for crowdsourcing of the information as this modality is in ubiquitous use. As a consequence, it was necessary to consider the effect on the habitability of an automotive voice interface system given a corpus of typed modality samples.

In chapter 4, methods for gathering command samples via a crowdsourcing approach were developed and the information generated by the study (both typed and spoken) was compared to understand the difference between typed and spoken commands. A typical example of the former is the command ’turn on right hand indicator‘ and the an example of the latter is ‘right indicator on‘. Word-counts and ambiguity of commands were assessed. Whilst differences were noted, the lack of an available command processor meant that it was not possible to identify the degree to which observed differences would affect the confidence with which a command processor could identify the action intended by a spoken command given typed or spoken corpus modalities.

Chapter 5 describes the development of a token-based natural language command processing method. This command processing method was shown to provide a reliable, sensitive and normalised confidence measure of whether a command is intended to enact an action by pattern matching against a corpus of previous commands for each possible action.

In this chapter, the command processing method developed in chapter 5 was used to examine the command samples obtained in study 3 (section 4.2) and compare the probabilities of correctly identifying an action intended by a spoken command using a corpus firstly of spoken command samples and secondly of typed command samples.

One possibility was that using a different modality between the corpus and the command would reduce system habitability in terms of the proportion of commands which could be identified (or system confidence in identifying the command). Consequences of a reduction in the ability of the system to identify an intended action could necessitate gathering an increased number of corpus samples (to mitigate the effect) or may reduce the habitability of the system below that required by users as noted in study 1 (chapter 3).

## 6.2 Method

Each of the set of samples obtained during study 3 was processed using the command processing method developed in chapter 5. Two measures were considered; firstly the proportion of samples for which the intended action could be immediately identified and secondly the proportion of samples where the intended action had the second highest confidence level and the two highest confidence levels were close such that they might result in a disambiguation action. This was repeated where firstly the corpus of samples commands was from a spoken modality (this test sequence is described in section 6.2.1) and secondly where the corpus of commands was in a typed modality (this test sequence is described in section 6.2.2).

### 6.2.1 Assessment of spoken commands using a spoken modality corpus

1. The set of spoken samples provided by one study participant was removed from the set of all spoken samples.
2. The set of spoken samples excluding the samples removed in (1) was used to generate the command processor corpus using the method described in section 5.5.
3. Each of the removed samples was assessed using the method described in section 5.6.
4. The removed samples were replaced in the corpus and samples for the next participant were removed.
5. Stages 2-4 were repeated until all samples were processed.

### 6.2.2 Assessment of spoken commands using a typed modality corpus

1. The set of typed samples provided by one study participant was removed from the set of typed samples.
2. The set of typed samples excluding the samples removed in stage 1 was used to generate the command processor corpus using the method described in section 5.5.
3. Each of the spoken samples from the participant whose typed samples were excluded from the corpus was assessed using the method described in section 5.6.
4. The removed samples were replaced in the corpus and samples for the next participant were removed.
5. Stages 2-4 were repeated until all samples were processed.

Study participants' typed samples were removed from the corpus before processing their spoken samples to ensure that the test conditions were directly comparable between the spoken and typed modality corpuses.

## 6.3 Results

Figures 6.1 to 6.8 display results for a matrix of the 12 actions and 20 participant samples obtained during study 3 (section 4.2). Figures 6.1 to 6.4 show the results for spoken commands, interpreted by a corpus of spoken commands using the process described in section 6.2.1. Figures 6.5 to 6.8 show similar results where spoken commands are interpreted using a typed corpus using the process described in section 6.2.2.

Figure 6.1 shows in green those commands for which the most probable calculated action identified was the intended action. The red markers show where the most probable calculated action was not the intended action.

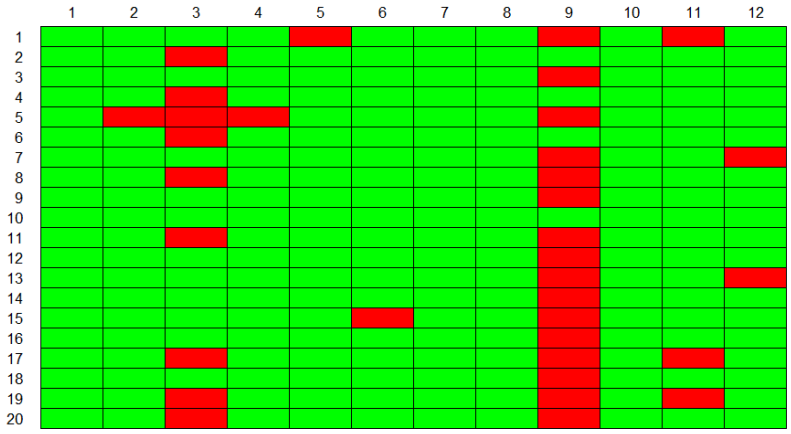



Figure 6.1. Map of commands correctly identified using a spoken corpus (study 4).

The intended action was identified in 206 of 240 commands, comprising an 85.8% success rate.

Figure 6.2 shows residual z-values (normalised confidence) obtained when comparing the most probable action for each command to a logarithmic model fitted to the other, less probable actions.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 10.53 | 158.21 | 60.78 | 144.65 | 1.97 | 155.87 | 25.84 | 176.46 | 1.43 | 19.54 | 1.25 | 3.65 |
| 2 | 10.47 | 152.47 | 1.46 | 20.45 | 5.85 | 127.46 | 25.74 | 5.45 | 33.05 | 14.18 | 1.86 | 5.08 |
| 3 | 19.01 | 134.23 | 1.61 | 226.85 | 39.27 | 1.55 | 37.74 | 55.33 | 1.43 | 18.44 | 1.27 | 5.40 |
| 4 | 18.92 | 132.19 | 1.48 | 142.47 | 13.83 | 5.58 | 4.95 | 42.84 | 46.91 | 13.79 | 2.17 | 4.33 |
| 5 | 8.73 | 0.44 | 1.42 | 2.54 | 3.34 | 149.14 | 13.52 | 8.43 | 2.04 | 25.67 | 1.16 | 4.35 |
| 6 | 19.17 | 134.14 | 1.41 | 255.58 | 192.64 | 5.28 | 209.31 | 5.74 | 11.38 | 18.15 | 1.81 | 4.38 |
| 7 | 14.13 | 14.98 | 1.56 | 42.95 | 26.25 | 127.41 | 12.40 | 8.39 | 0.46 | 19.13 | 2.04 | 1.27 |
| 8 | 24.09 | 121.58 | 1.41 | 216.26 | 30.61 | 157.50 | 98.93 | 8.31 | 0.97 | 14.16 | 2.08 | 4.37 |
| 9 | 16.60 | 122.53 | 50.11 | 128.70 | 325.34 | 3.07 | 170.50 | 8.55 | 2.07 | 11.55 | 1.64 | 1.01 |
| 10 | 16.88 | 132.84 | 56.29 | 227.14 | 35.00 | 2.91 | 5.65 | 5.73 | 1.51 | 13.80 | 1.91 | 1.18 |
| 11 | 19.22 | 136.81 | 1.28 | 221.65 | 50.85 | 6.14 | 6.24 | 7.57 | 1.81 | 18.50 | 1.80 | 4.36 |
| 12 | 16.95 | 3.20 | 1.67 | 242.58 | 40.02 | 4.80 | 143.37 | 9.93 | 0.46 | 9.87 | 1.14 | 4.29 |
| 13 | 19.07 | 125.37 | 1.64 | 123.36 | 118.88 | 111.47 | 18.71 | 5.80 | 2.36 | 29.32 | 2.20 | 1.58 |
| 14 | 22.81 | 300.28 | 7.13 | 382.12 | 1.28 | 222.31 | 5.40 | 6.83 | 1.99 | 17.38 | 2.51 | 1.37 |
| 15 | 10.61 | 135.61 | 60.07 | 254.03 | 317.93 | 3.23 | 23.60 | 5.42 | 2.37 | 35.45 | 1.87 | 5.10 |
| 16 | 27.07 | 143.13 | 1.52 | 234.36 | 229.58 | 4.18 | 30.34 | 6.11 | 1.63 | 21.19 | 2.00 | 4.27 |
| 17 | 4.07 | 123.10 | 1.80 | 5.28 | 40.82 | 118.44 | 3.41 | 2.63 | 0.69 | 16.91 | 2.03 | 2.68 |
| 18 | 17.18 | 2.71 | 1.61 | 201.02 | 18.02 | 100.68 | 132.30 | 5.82 | 0.65 | 13.89 | 1.85 | 6.83 |
| 19 | 7.97 | 2.77 | 1.18 | 1.30 | 309.44 | 90.28 | 5.20 | 63.51 | 0.18 | 37.86 | 2.62 | 1.45 |
| 20 | 15.55 | 144.80 | 1.93 | 280.39 | 20.40 | 171.95 | 130.18 | 9.46 | -0.13 | 13.87 | 2.33 | 5.94 |

| Range | |
|---|---|
| 4 | 999 |
| 3 | 4 |
| 2.5 | 3 |
| 2 | 2.5 |
| 1.5 | 2 |
| 1 | 1.5 |
| 0.5 | 1 |
| 0 | 0.5 |
| -0.5 | 0 |

Colour

Figure 6.2. Map of normalised confidence levels for the highest confidence action using a spoken corpus (study 4).

Where the action with the highest level of pattern matching was not the intended action, the action with the second highest level of pattern matching was considered to identify whether it was the one intended by the command. If this was the case and it was possible to identify potential ambiguity, including the two most probable actions in a disambiguation interaction with the user would include the intended action. Figure 6.3 shows in green those commands for which the second most probable action identified was the intended action.

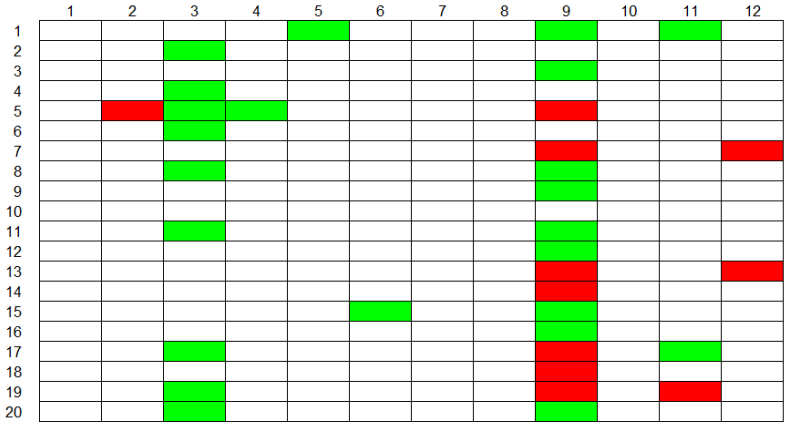



Figure 6.3. Map of commands correctly identified by the second highest-confidence action using a spoken corpus (study 4).

Of the 34 commands which were not correctly matched by action with the highest level of pattern matching, 23 commands were correctly identified by the action with the second highest level of pattern matching. This results in 95.8% of commands either correctly identified by the two most probable calculated actions.

Figure 6.4 shows the residual values obtained when comparing the action with the second highest level of pattern matching for each command to a logarithmic model fitted to actions with lower levels of pattern matching.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.16 | -0.15 | -0.13 | -0.14 | 0.97 | -0.14 | -0.02 | -0.15 | 1.47 | -0.08 | 1.31 | 0.48 |
| 2 | 0.16 | -0.15 | 1.40 | 0.01 | 0.32 | -0.15 | -0.02 | 0.21 | -0.10 | 0.03 | 0.67 | -0.08 |
| 3 | 0.03 | -0.15 | 0.64 | -0.15 | -0.09 | 1.52 | -0.07 | -0.10 | 1.48 | -0.01 | 0.63 | -0.13 |
| 4 | 0.03 | -0.15 | 1.42 | -0.14 | 0.10 | 0.45 | 0.52 | -0.08 | -0.12 | 0.03 | 0.94 | 0.28 |
| 5 | 0.23 | 0.79 | 1.22 | 1.01 | 0.75 | -0.15 | 0.10 | 0.10 | 0.51 | -0.02 | 0.54 | 0.35 |
| 6 | 0.03 | -0.15 | 1.20 | -0.15 | -0.15 | 0.48 | -0.16 | 0.23 | 0.04 | -0.01 | 0.70 | 0.33 |
| 7 | 0.09 | -0.05 | 0.83 | -0.17 | -0.06 | -0.15 | 0.13 | 0.04 | 1.01 | -0.01 | 1.20 | 0.92 |
| 8 | -0.01 | -0.15 | 1.23 | -0.15 | -0.07 | -0.14 | -0.14 | 0.08 | 1.02 | 0.03 | 0.64 | 0.34 |
| 9 | 0.06 | -0.15 | -0.12 | -0.14 | -0.16 | 0.84 | -0.15 | 0.08 | 1.20 | 0.11 | 0.64 | 0.47 |
| 10 | 0.05 | -0.15 | -0.13 | -0.15 | -0.10 | 0.89 | 0.44 | 0.24 | 1.55 | 0.04 | 0.70 | 0.65 |
| 11 | 0.03 | -0.15 | 1.59 | -0.15 | -0.09 | 0.39 | 0.39 | 0.19 | 1.34 | -0.01 | 0.71 | 0.32 |
| 12 | 0.05 | 0.22 | 0.70 | -0.15 | -0.09 | 0.53 | -0.15 | 0.07 | 0.86 | 0.11 | 1.46 | 0.25 |
| 13 | 0.03 | -0.15 | 0.63 | -0.14 | -0.15 | -0.15 | 0.03 | 0.23 | 0.78 | -0.10 | 0.93 | 1.15 |
| 14 | 0.00 | -0.16 | 0.09 | -0.16 | 0.86 | -0.15 | 0.46 | 0.21 | 0.82 | -0.03 | 0.03 | 1.56 |
| 15 | 0.17 | -0.15 | -0.12 | -0.16 | -0.16 | 0.80 | -0.01 | 0.21 | 0.93 | -0.11 | 0.67 | -0.08 |
| 16 | -0.04 | -0.15 | 0.73 | -0.15 | -0.16 | 0.62 | -0.04 | 0.21 | 1.16 | -0.08 | 0.79 | 0.32 |
| 17 | 0.64 | -0.15 | 0.73 | 0.48 | -0.09 | -0.14 | 0.77 | 0.90 | 0.59 | 0.01 | 0.69 | 0.04 |
| 18 | 0.05 | 0.22 | 0.63 | -0.15 | 0.04 | -0.15 | -0.15 | 0.17 | 1.25 | 0.03 | 0.63 | 0.05 |
| 19 | 0.26 | 0.09 | 1.24 | 1.57 | -0.16 | -0.13 | 0.48 | -0.11 | 0.10 | -0.08 | 0.71 | 1.03 |
| 20 | 0.06 | -0.15 | 1.12 | -0.15 | -0.01 | -0.15 | -0.14 | 0.09 | 0.25 | 0.04 | 0.89 | 0.09 |

| Range | |
|---|---|
| 1.5 | 999 |
| 1.25 | 1.5 |
| 1 | 1.25 |
| 0.75 | 1 |
| 0.5 | 0.75 |
| 0.25 | 0.5 |
| 0 | 0.25 |
| -0.25 | 0 |
| -0.5 | -0.25 |

Colour

Figure 6.4. Map of normalised confidence levels for the second-highest confidence action using a spoken corpus (study 4).

Figure 6.5 shows a matrix similar to Figure 6.1 where a typed corpus has been used to interpret the command.

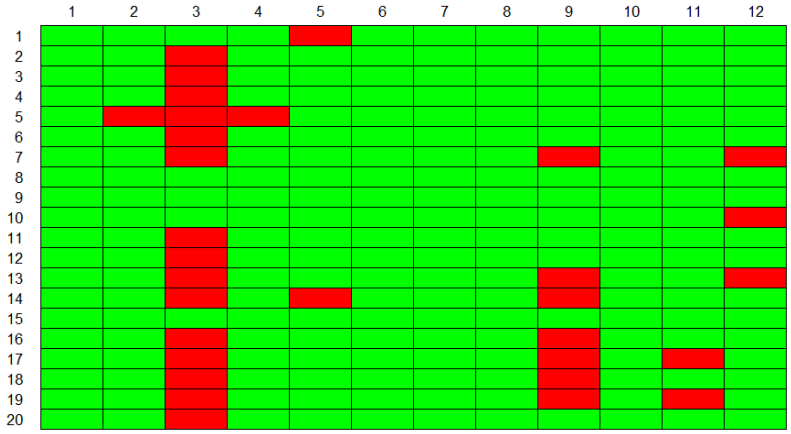



Figure 6.5. Map of commands correctly identified using a typed corpus (study 4).

The intended action was identified in 209 of 240 commands, comprising an 87.1% success rate.

Figure 6.6 shows a matrix similar to Figure 6.3.2 where a typed corpus has been used to generate residual values.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 16.20 | 14.55 | 2.65 | 11.28 | 0.80 | 55.63 | 8.46 | 15.90 | 3.25 | 12.24 | 2.45 | 3.76 |
| 2 | 18.53 | 12.70 | 3.63 | 12.33 | 9.54 | 65.68 | 6.61 | 5.00 | 3.70 | 7.81 | 4.04 | 4.50 |
| 3 | 22.11 | 9.45 | 0.71 | 12.89 | 23.29 | 5.32 | 12.96 | 15.50 | 3.77 | 9.94 | 3.61 | 4.35 |
| 4 | 22.54 | 10.86 | 2.34 | 10.08 | 9.76 | 18.62 | 4.34 | 15.60 | 5.40 | 9.26 | 5.60 | 3.70 |
| 5 | 17.42 | -0.12 | 3.00 | 1.80 | 4.61 | 35.44 | 4.47 | 5.98 | 0.60 | 8.98 | 6.42 | 7.25 |
| 6 | 25.28 | 7.11 | 4.91 | 11.95 | 10.22 | 19.16 | 23.87 | 4.77 | 0.54 | 16.34 | 2.55 | 9.95 |
| 7 | 23.51 | 5.52 | 0.73 | 13.12 | 20.51 | 38.77 | 7.23 | 6.18 | 1.22 | 9.94 | 3.61 | 1.70 |
| 8 | 25.01 | 9.12 | 0.65 | 13.99 | 15.62 | 15.35 | 15.51 | 7.33 | 3.17 | 6.53 | 12.54 | 2.95 |
| 9 | 19.25 | 10.84 | 1.80 | 8.57 | 26.47 | 11.03 | 35.24 | 6.63 | 1.47 | 10.59 | 3.05 | 1.36 |
| 10 | 22.90 | 11.75 | 2.30 | 17.39 | 30.51 | 12.44 | 4.37 | 7.11 | 2.68 | 7.50 | 5.76 | 2.01 |
| 11 | 19.49 | 7.11 | 1.57 | 11.13 | 19.85 | 16.54 | 6.20 | 9.08 | 1.59 | 11.49 | 5.80 | 3.71 |
| 12 | 29.40 | 2.85 | 1.31 | 22.51 | 29.21 | 17.58 | 34.22 | 5.35 | 1.52 | 5.60 | 2.79 | 10.95 |
| 13 | 22.32 | 10.71 | 0.89 | 6.69 | 8.09 | 43.37 | 5.74 | 7.03 | 2.54 | 16.99 | 7.00 | |
| 14 | 22.29 | 9.24 | 1.08 | 13.51 | 0.99 | 41.46 | 5.34 | 6.55 | 3.77 | 11.21 | 4.08 | 11.83 |
| 15 | 13.96 | 9.21 | 1.69 | 13.73 | 21.33 | 2.78 | 6.06 | 4.77 | 3.08 | 13.96 | 2.55 | 7.48 |
| 16 | 30.17 | 11.16 | 1.50 | 10.91 | 12.82 | 12.80 | 10.37 | 3.26 | 1.06 | 23.30 | 4.04 | 12.40 |
| 17 | 5.35 | 14.27 | 3.35 | 6.62 | 34.92 | 51.08 | 2.63 | 2.81 | 0.44 | 10.85 | 1.27 | 1.93 |
| 18 | 22.73 | 1.74 | 0.86 | 17.45 | 15.02 | 54.05 | 34.26 | 5.18 | 0.73 | 7.23 | 4.30 | 7.35 |
| 19 | 35.70 | 1.11 | 3.24 | | 19.75 | 36.55 | 1.85 | 20.77 | 0.38 | 24.49 | 2.91 | 3.30 |
| 20 | 39.41 | 12.07 | 3.27 | 18.61 | 41.65 | 55.83 | 31.94 | 4.76 | 1.47 | 7.27 | 6.07 | 10.81 |

| Range | |
|---|---|
| 4 | 999 |
| 3 | 4 |
| 2.5 | 3 |
| 2 | 2.5 |
| 1.5 | 2 |
| 1 | 1.5 |
| 0.5 | 1 |
| 0 | 0.5 |
| -0.5 | 0 |

Colour

Figure 6.6. Map of normalised confidence levels for the highest confidence action using a typed corpus (study 4).

Figure 6.7 shows a matrix similar to Figure 6.3 where a typed corpus has been used to consider the degree to which actions not matched to the action with the highest level of pattern matching can instead be matched to the action with the second highest level of pattern matching.

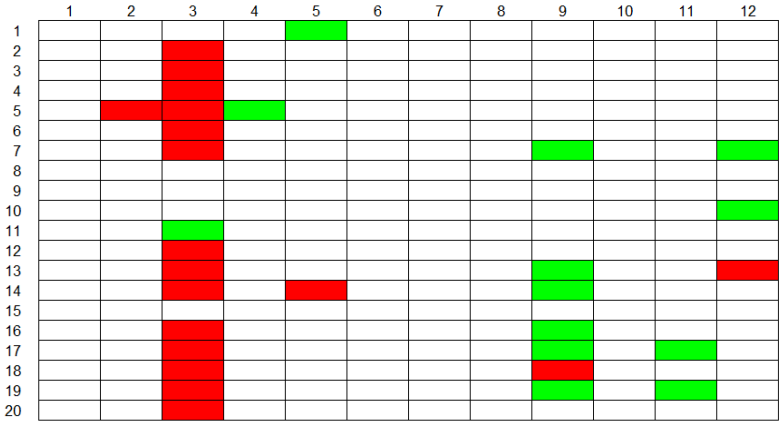


Figure 6.7. Map of commands correctly identified by the second highest-confidence action using a typed corpus (study 4).

Of the 31 commands which were not correctly matched by the action with the highest level of pattern matching, 13 commands were correctly identified by the action with the second highest level of pattern matching. This results in 92.5% correctly identified by the two actions with the highest level of pattern matching.

Figure 6.8 shows a matrix similar to Figure 6.4 where a typed corpus has been used to interpret the command.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.00 | -0.25 | -0.52 | -0.27 | 0.88 | -0.18 | 0.25 | -0.24 | 0.67 | 0.03 | 0.81 | 0.60 |
| 2 | -0.03 | -0.26 | 0.35 | -0.08 | 0.05 | -0.17 | 0.35 | 0.41 | -0.46 | 0.24 | 0.56 | 0.52 |
| 3 | -0.02 | -0.30 | 1.07 | -0.26 | -0.06 | 0.42 | 0.11 | -0.22 | 0.53 | 0.16 | 0.36 | 0.54 |
| 4 | -0.02 | -0.28 | 0.46 | -0.29 | 0.11 | 0.02 | 0.59 | -0.22 | -0.38 | 0.20 | 0.32 | 0.50 |
| 5 | -0.04 | 0.28 | -0.28 | 0.90 | 0.10 | -0.19 | 0.54 | 0.06 | 1.17 | 0.18 | 0.13 | -0.18 |
| 6 | -0.06 | -0.33 | 0.10 | -0.27 | -0.28 | -0.01 | -0.20 | 0.48 | -0.70 | -0.01 | 0.87 | 0.02 |
| 7 | -0.03 | -0.18 | 1.39 | -0.26 | -0.05 | -0.19 | 0.32 | 0.19 | 1.35 | 0.16 | 0.62 | 0.76 |
| 8 | -0.09 | -0.27 | 1.17 | -0.23 | -0.06 | -0.22 | -0.13 | -0.25 | -0.03 | 0.34 | 0.04 | 0.26 |
| 9 | 0.01 | -0.25 | -0.52 | -0.28 | -0.20 | 0.11 | -0.10 | 0.19 | 1.30 | 0.04 | 0.75 | 1.42 |
| 10 | -0.02 | -0.27 | -0.59 | -0.24 | -0.10 | 0.11 | 0.59 | 0.16 | 0.87 | 0.28 | 0.30 | 0.99 |
| 11 | -0.01 | -0.33 | 0.52 | -0.27 | -0.15 | -0.01 | 0.39 | 0.11 | 1.13 | 0.13 | 0.27 | 0.49 |
| 12 | -0.05 | 0.51 | 0.37 | -0.22 | -0.06 | 0.03 | -0.08 | 0.35 | 1.41 | 0.36 | 0.55 | -0.12 |
| 13 | -0.03 | -0.29 | 0.96 | -0.37 | -0.32 | -0.17 | 0.40 | 0.19 | 0.76 | 0.00 | 0.17 | -0.63 |
| 14 | -0.03 | -0.30 | 0.95 | -0.26 | 0.83 | -0.19 | 0.47 | 0.19 | 0.16 | 0.12 | 0.55 | -0.05 |
| 15 | 0.06 | -0.30 | -0.64 | -0.26 | -0.22 | 0.85 | 0.40 | 0.50 | 0.34 | -0.02 | 0.91 | 0.22 |
| 16 | -0.13 | -0.27 | 0.27 | -0.28 | -0.26 | 0.10 | 0.17 | 0.67 | 1.29 | -0.12 | 0.59 | -0.16 |
| 17 | 0.43 | -0.25 | 0.34 | 0.32 | -0.10 | -0.19 | 0.98 | 0.86 | 0.33 | 0.15 | 1.50 | 1.10 |
| 18 | -0.02 | 0.50 | 1.13 | -0.24 | 0.03 | -0.17 | -0.07 | 0.37 | 0.95 | 0.27 | 0.51 | 0.21 |
| 19 | -0.09 | 0.61 | 0.35 | -0.56 | -0.23 | -0.20 | 1.23 | -0.22 | 1.11 | -0.09 | 0.44 | 0.23 |
| 20 | -0.12 | -0.27 | 0.30 | -0.23 | -0.14 | -0.19 | -0.06 | 0.24 | 0.53 | 0.27 | 0.22 | -0.13 |

| Range | | Colour |
|---|---|---|
| 1.25 | 1.5 | |
| 1 | 1.25 | |
| 0.75 | 1 | |
| 0.5 | 0.75 | |
| 0.25 | 0.5 | |
| 0 | 0.25 | |
| -0.25 | 0 | |
| -0.5 | -0.25 | |
| -1 | -0.5 | |

Figure 6.8. Map of normalised confidence levels for the second-highest confidence action using a typed corpus (study 4).

The number of commands for which the action with the highest level of pattern matching was the intended action was compared between the spoken corpus and the typed corpus using a two proportions test, no significant difference was found [$z = 0.4$, $p = 0.689$].

Where the action with the highest level of pattern matching was not the intended action, the number of commands where the action with the second highest level of pattern matching was the one intended were compared between the two corpus modalities using a two proportions test. A significant difference was identified with the spoken corpus more able to interpret the second highest level of pattern matching [$z = 2.15$, $p = 0.031$].

## 6.4 Discussion

The percentage of commands for which the intended action was identified by the highest level of pattern matching was 85.8% for the spoken corpus and 87.1% for the typed corpus, a level of habitability which was slightly below that required by participants in Study 1 (chapter 3). A two-proportions test did not find a significant difference between these results. Comparison of figures 6.3.1 and 6.3.5 highlighted differences in action identification:

- Likelihood of identifying the intended action was action dependent. Actions 3 and 9 were inherently ambiguous and were less likely to be correctly identified. Examination of the two actions showed a number of common words used in commands for these actions.
- The spoken corpus had more difficulty in identifying action 9, the typed corpus had more difficulty in identifying action 3.
- For the typed corpus, action 3 was particularly problematic with 14 of the 20 commands not identifying the intended command as either the highest or second highest level of pattern matching.
- It can be noted from tables 6.1 and 6.2 that commonality of the word 'on' may be a particular factor in the difficulty of the system to clearly differentiate the commands and this could be mitigated by a method of separately handling data (such as the state of on or off).

Table 6.1, typed commands for action 3 and action 9

| **Action 3** | **Action 9** |
|---|---|
| Hazards on | Illuminate speedo |
| emergency lights on | raise display lighting |
| Hazards on | Instrument pack illumination up |
| hazard lights | increase instrument panel lighting |
| hazard warning lights on | |
| turn hazard lights on | light up display |
| hazard warning on | |
| hazard warning | dials red |
| put on hazard lights | brighten the dial illumination |
| hazards on | display lighter |
| Hazards on | bright dials |
| Hazard Warning Lights On | Increase Cluster Illumination |
| hazards on | display light on |
| put hazards on | light up dashboard |
| hazards | panel lights up |
| hazards on | illuminate dials |
| hazard warning lights | illuminate dials |
| flash hazards | increase brightness of dials |
| turn on hazard lights | increase instrument lighting |
| hazard lights on | light up screen |
| hazard lights on | brighten dashboard display lights |

Table 6.2, spoken commands for action 3 and action 9

| **Action 3** | **Action 9** |
|---|---|
| Hazards on | Wipers on |
| Hazard warning lights | Front wipers on |
| Warning lights | windscreen wipers intermittent |
| Turn hazards on | Windscreen wipers on |

| | |
|---|---|
| Warning lights | wipers on |
| Hazards on | Switch wipers on |
| Hazards on | Windscreen wipers intermittent |
| Hazard lights on | Windscreen wipers on |
| Small lights on | Window wipers on slow continuous |
| Hazard lights on | Wipers on |
| Hazards | Wipers on |
| Hazard lights on | Turn on windscreen wipers |
| Hazards | Windscreen wipers on |
| Hazards | Wipers on |
| Hazard lights on | Windscreen wipers on slow |
| Hazards on | Wipers intermittent |
| Hazard lights please | Windscreen wipers on |
| turn on hazard warning lights | turn on windscreen wipers |
| put hazards on | turn the windscreen wipers on |
| switch hazards on | Turn on windscreen wipers intermittent |
| hazard lights on | wipers on |
| Hazards | Wipers intermittent |
| hazards on | wipers on |
| Hazards | Put the windscreen wipers about medium |
| Front lights on | Windscreen wipers |
| Activate hazard lights | Activate windscreen wipers on slow |

When identifying the intended action from the highest level of pattern matching, the two modalities demonstrated comparable performance. Corpus modality was shown to affect which actions have inherent identification difficulty.

Figures 6.3.2 and 6.3.6 show the residual values used to identify the intended action. In each case, it can be seen from the colour-coded levels that lower residuals tend to correspond with incorrect action identification. This suggests that a threshold of confidence might be identified, above which it may be assumed that the action with the highest level of pattern matching is the one intended.

It can also be seen from Figures 6.3.2 and 6.3.6 that in some cases where an incorrect action was identified, the residual value was higher than other cases where the correct action was identified. This would therefore require that a threshold for taking an action would either include some incorrect actions or would exclude some correct actions. The level which would give the best compromise in terms of user satisfaction cannot be directly determined from this result.

Recovery of an incorrectly identified action requires two calculations:

1. That there is insufficient confidence to immediately enact the action must first be identified (using in this case, the threshold described in the previous paragraph)
2. Identification of the two ambiguous actions such that one of these is the required action.

The results shown in figures 6.3.3 and 6.3.7 show the chance of identifying two candidate actions where one is the action intended. The two-sample t-test performed showed that the spoken corpus was statistically more likely to identify the intended action using the second highest residual value. However this can be seen to be dependent on action 3 where the typed corpus correctly identified the intended action from action with the second highest level of pattern matching in only one of the fifteen commands.

Figures 6.3.4 and 6.3.8 show the residual levels used to identify action with the second highest level of pattern matching. Unlike the case of action with the highest level of pattern matching (figures 6.3.2 and 6.3.6), no visually clear link is shown between the residual level and the likelihood of identifying the intended action.

## 6.5 Conclusion

In the introduction to this chapter (section 6.1) the potential was raised that using a different modality between a voice interface command-sample corpus and the spoken command could reduce the ability of the system to pattern-match commands against corpus samples and necessitate gathering an increased number of samples (to mitigate the effect) or may reduce the habitability of the system below that required by users as noted in study 1 (chapter 3). This study has determined that (in conjunction with the crowdsourcing approach developed in chapter 4 and the natural language command processing method described in chapter 5), use of typed modality command samples in place of spoken modality samples does not reduce the habitability of the system in terms of the proportion of spoken commands correctly assigned to actions (research question 2 & 4). Therefore, the number of corpus samples needed to provide a given degree of action-identification confidence is not impacted by the use of typed modality corpus samples.

The only observed area in which a typed modality corpus was less able than a spoken modality corpus was in identifying an intended action using the second-most probable action (i.e. via a disambiguation action). This effect would only occur on the occasions when the intended action could not be immediately identified.

Whilst results support the use of a typed modality corpus, they also show that the thresholds for types of user-response are unclear as any threshold will be a compromise between acting on incorrect results and implementing recovery methods for properly matched actions. Further study may be needed to identify optimal thresholds for user responses and this is progressed in chapter 8.

One limitation of the above study is that it assumes that commands are correctly recognised by the voice recognition system. In cases where one or more words cannot be correctly recognised, the effect of corpus modality is not clear. In study 5 (chapter 7), the command

processing method described in chapter 5 is used to assess commands from study 3 with words removed, simulating the effect of voice recognition errors.

# *Chapter 7: Robustness of command identification to voice recognition errors (study 5)*

## 7.1 Introduction

### Aim of chapter 7

Research question 4 (section 1.4) addresses the robustness of the command processing method developed in chapter 5 in combination with crowdsourced, typed modality samples to partially-recognised command phrases.

### Overview

Study 4 (Chapter 6) demonstrated that in conjunction with the natural language token-based command processor developed in chapter 5, use of a crowdsourced, typed modality corpus is similarly effective to a spoken corpus when identifying the action intended by a spoken command. However, in an automotive environment, there are several noise-factors which may prevent the spoken command phrase being properly interpreted into the intended words by the voice recognition system. These include:

- The microphone may be further away compared to other voice interfaces.
- The vehicle interior acoustics and microphone positioning may not be conducive to clear recording.
- Driving stress may cause changes to intonation.
- Ambient noise (for example a passing vehicle) may obscure speech.

For these reasons, automotive voice interfaces need to be particularly robust to noise factors which result in partially understood command phrases. One possibility is that habitability (in terms of proportion of intended actions identified from spoken commands) could be reduced more when using a typed modality corpus than when using a spoken modality corpus.

The study described in this chapter considers the ability of the method developed in chapter 5 to interpret spoken commands obtained in study 3 (chapter 4.2). Commands with words removed (thereby simulating incorrect voice recognition) are put to the command processing method and the degree to which this affects command identification for both typed and spoken corpuses is examined.

## 7.2 Method

The set of commands obtained from study 3 (section 4.2) was used for this study. To create the set of simulated incorrectly recognised commands, the set of spoken commands was processed by omitting each command word in turn. For example, the command ‘open sun roof’ given by a participant in study 3 would result in a set of incorrectly recognised commands comprising ‘sun roof’, ‘open roof’ and ‘open sun’. Commands from the set of data

obtained in study 3 varied in word count and consequently the process of simulating incorrectly recognised commands generated a different number of commands for each action. In total, 746 incorrectly recognised commands were generated. Table 7.1 shows three commands expanded in this manner.

Table 7.1, three commands from study 3 with a single missing word.

| Command | Roof open | Reset trip computer | Adjust steering wheel down |
|---|---|---|---|
| Omitted word 1 | Open | Trip computer | Steering wheel down |
| Omitted word 2 | Roof | Reset computer | Adjust wheel down |
| Omitted word 3 | | Reset trip | Adjust steering down |
| Omitted word 4 | | | Adjust steering wheel |

The percentage of incorrectly recognised commands for which the intended action was correctly identified was calculated firstly for a corpus based on a typed modality and secondly a spoken modality corpus. The method described in section 5.5 was used to generate each corpus using the data obtained from study 3.

## 7.3 Results

For each modality, the percentage of commands for which the most probable calculated action was the action intended is shown in Figure 7.1.

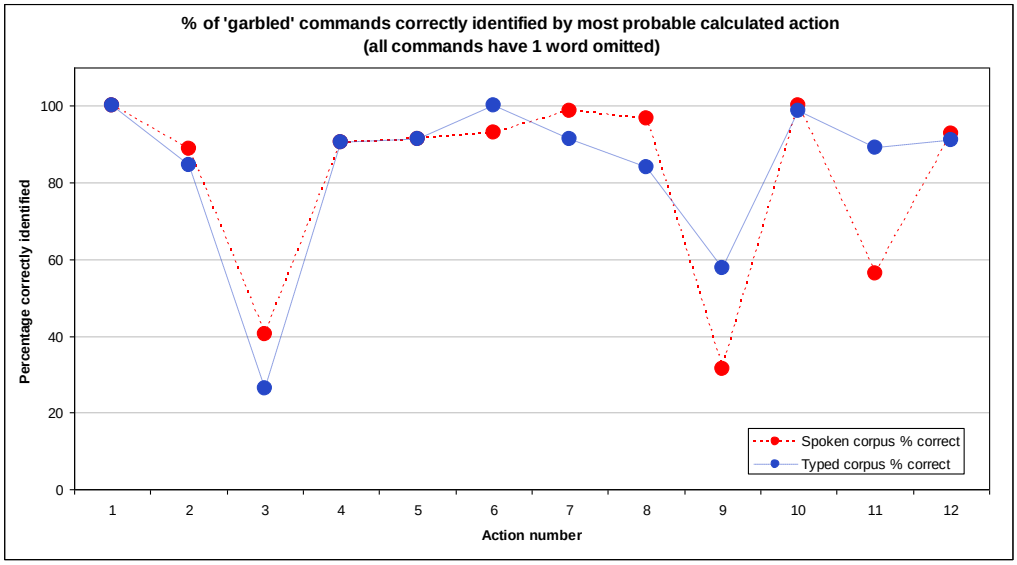



Figure 7.1. Percentage of correctly identified actions for typed and spoken modalities (study 5).

To consider the potential for a disambiguation action to identify the intended action, the percentage of commands where the two most probable calculated actions included the intended action was calculated. The result can be seen in Figure 7.2.

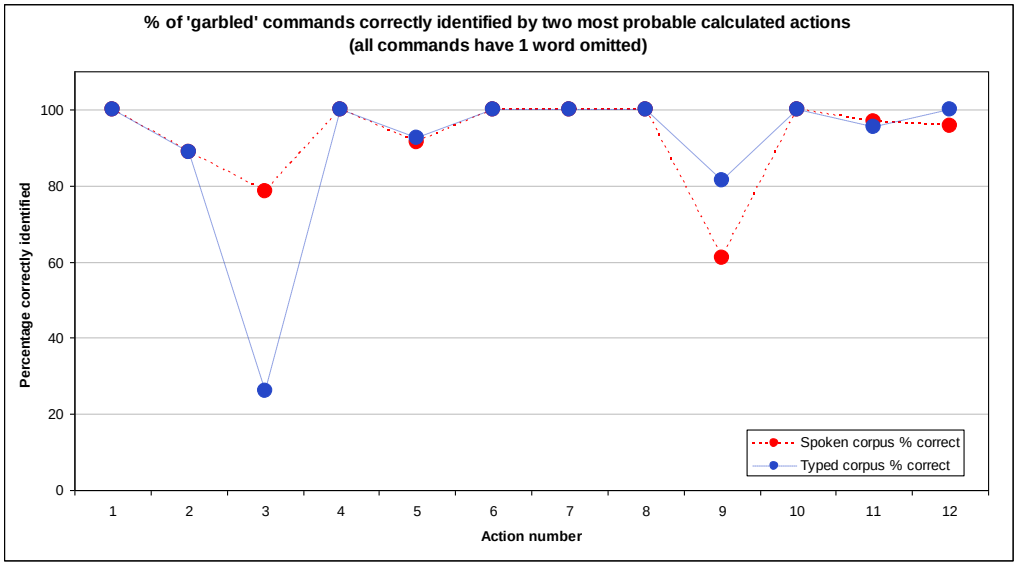



Figure 7.2. Percentage of commands for which one of the two highest confidence actions was correct for typed and spoken modalities (study 5).

## 7.4 Discussion

For actions other than 3 and 9, the percentage of commands correctly identified for both modalities was between 90% and 100%. 90% recognition suggests that the method developed in chapter 5 is robust to one-word command recognition errors and that this is the case irrespective of whether the command processor corpus comprises a typed or spoken modality. For ten of the twelve actions, results suggest that this method would also respond well to lexical selection errors (replacing words with other of similar but contextually incorrect meaning) as described by Eysenck and Keane [73, p. 366].

For commands 3 and 9, the degree of ambiguity and ability of the method to identify the intended action remains at a level similar to that identified in chapter 6. Simulating a damaged command by omitting words has not therefore significantly affected command processing performance for either modality. It was expected that omission of a word from commands would result in a reduction in the ability of the command processing method to identify intended actions. That this did not occur demonstrated a surprising degree of robustness of the method to issues with command recognition.

It was possible that the command processing method would be similar in performance to a basic key-word matching method (whilst still being enhanced by use of a corpus of samples from participants whose choice of command phrasing is similar to that of a potential user), i.e. that only specific command words such as nouns and/or verbs would affect the degree of pattern-matching. If this were the case, when the words which might be selected as 'key' were omitted, commands would not be recognised. Since there was no reduction in proportion of recognised commands irrespective of the word omitted, this shows that matching all command words provides a degree of robustness to word omissions.

## 7.5 Conclusion

The introduction to this chapter (section 7.1) noted that an automotive environment is challenging for voice recognition and identified a number of factors which may result in one of the words spoken in the command phrase being incorrectly recognised. The potential was noted that whilst habitability (in terms of proportion of intended actions identified from spoken commands) may reduce in all cases, it may be more adversely affected when using a typed-modality corpus.

The results of this study have shown that the approach of using command samples and the token-based natural language command processor developed in chapter 5 is robust to single-word voice recognition errors (research question 4). It has also demonstrated that this degree of robustness is similar for both a typed modality corpus and a spoken-modality corpus.

The results obtained from the study in this chapter and study 4 (chapter 6) suggest that a crowdsourced, typed modality corpus is a robust approach for processing of spoken-commands and this supports the use of crowdsourcing of typed modality command samples as a practical and efficient approach to obtaining a distributed data set.

In study 6 (chapter 8), the prompting method and typed corpus obtained from study 3 and the command processing method developed in chapter 5 were used to consider optimal action-identification confidence thresholds for selecting different types of system response and the degree to which fixed thresholds meet the needs of a population of potential users.

# *Chapter 8: Selecting optimal system responses by confidence-level thresholds (study 6).*

## 8.1 Introduction

### Aim of chapter 8

Research question 5 (section 1.4) relates to the potential to compare confidence in identifying an action to fixed response thresholds to select between different types of response to users. Study 6 aims to assess the use of fixed response thresholds and also investigates the potential to identify optimal response threshold levels using a crowdsourcing approach.

### Overview

Baber noted that "Habitable speech systems are preferred by users" [5, p.660]. Studies 2-5 have focussed on whether the proposed, crowdsourced, typed modality methods for gathering command samples provide a comparable level of habitability (including both reliability and robustness to noise factors) when compared to spoken command samples.

An additional factor which will affect user satisfaction is the manner in which a system selects a response to users in the circumstances where an intended action cannot be identified with sufficient confidence to immediately enact. Ideally, a system response should match the response that users would prefer to receive in the context of that level of confidence. This flow of information from computer to human differs from habitability in that the human is likely to understand whichever response is given but is similar in that user satisfaction may relate to the match between the responses that the system employs and those which the user considers appropriate given the context. For example, if the user coughs and the computer asks whether to take a specific action, the user is unlikely to be satisfied. Similarly if the user gives a very precise and clearly worded command, a response asking for confirmation is likely to be less satisfying to a user than simply taking the action. In an ideal case, the normalised confidence levels for action identification derived by the command processing method developed in chapter 5 could be compared to a single set of threshold values to select an response (for example, confirm before acting) which aligns with the preference of all users. However, one possibility is that different users may require different response thresholds. This could necessitate user adjustment, learning algorithms or some form of user categorisation to achieve effective threshold levels.

If it were necessary to establish different response thresholds for different languages and/or dialects, the approach of crowdsourcing command samples would only avoid scaling of resources if the information needed for identifying threshold levels could also be crowdsourced. The study described in this chapter also examines whether this is possible. To gather threshold data, it would be necessary to imbue study participants with sufficient

understanding of the function of threshold adjustment. One challenge is that users could entirely fail to understand the function and be unable to identify threshold levels or communicating the functions may take sufficient effort that users may not be willing to undergo such training.

In this study, participants were presented with an interface requiring them to issue typed commands in response to the animated prompts used in study 3 (section 4.2). The command processing method developed in chapter 5 was then used to identify a confidence percentage for each action and these were translated into system and user-interface responses by comparing confidence levels to three response-thresholds (described in section 8.2.3). Command processor method responses were displayed to the participants who were then asked to change the thresholds to optimise responses according to their preference.

In addition to investigating the effect of modality on command processor system responses, the information obtained during study 6 enabled the following factors to be investigated:

- The mean thresholds chosen by participants
- The similarity of the mean threshold chosen by participants to those estimated in study 1.
- Degree of fit of participant response to possible parametric models
- Whether participant responses are single or multimodal
- Whether threshold levels are related by individual participants
- The degree to which threshold models overlap
- Given the threshold model overlap, the degree to which optimal threshold settings may satisfy a population of users.

## 8.2 Method

### 8.2.1 Study interface design and operation

This study was designed to be completed by a participant without a researcher present in order to evaluate the potential to employ a crowdsourcing approach. The tool was built into a Microsoft Excel workbook with four study stages:

0. Introduction to the study and the study consent form
1. Capture of participants' typed commands in response to animated vehicle-action prompts.
2. Display of command processing system responses to participants' commands and response-threshold controls to allow participants to affect the responses.
3. Capture of participant information for response sub-grouping.

The consent form allowed participants to agree to the conditions stated and proceed with the study or advised participants to close the workbook. The date and time was recorded when consent to proceed was given. Participants who consented to continue with the study then proceeded to stage 1 where they were presented a text prompt instructing them to press the first of twelve numbered buttons displayed. This displayed an animation of a vehicle action taking place; the twelve animations were taken from study 3. An example is shown in Figure 4.11. Having viewed the animated action, participants were instructed to decide on the command they would use to enact the action and type this in the adjacent text field. Participants were asked to proceed sequentially until all twelve text fields were filled and then proceed to stage 2 of the study.

In stage 2 of the study, participants were shown the twelve commands they entered in stage 1 of the study with command processing method responses appended to them. Participants were also shown an additional twelve pre-set commands for the same prompts also with the command processing method responses appended; these were selected to cause ambiguity and poor action identification and ensure that participants considered such cases. A slider allowed participants to move one of the response thresholds. The interface used for stage 2 is shown in Figure 8.2. Each slider was introduced individually before allowing participants to use all three simultaneously to prevent participants becoming confused by range of available options.

System responses were constrained so that only information relevant to the threshold available for tuning was presented. Figure 8.2 shows command processing method responses which are limited to a statement of which action was taken or a response that no action was immediately taken. The circled command is an example of an incorrect action being taken given the 90% level of confidence required to enact a command. This level was set by the participant using the slider shown.

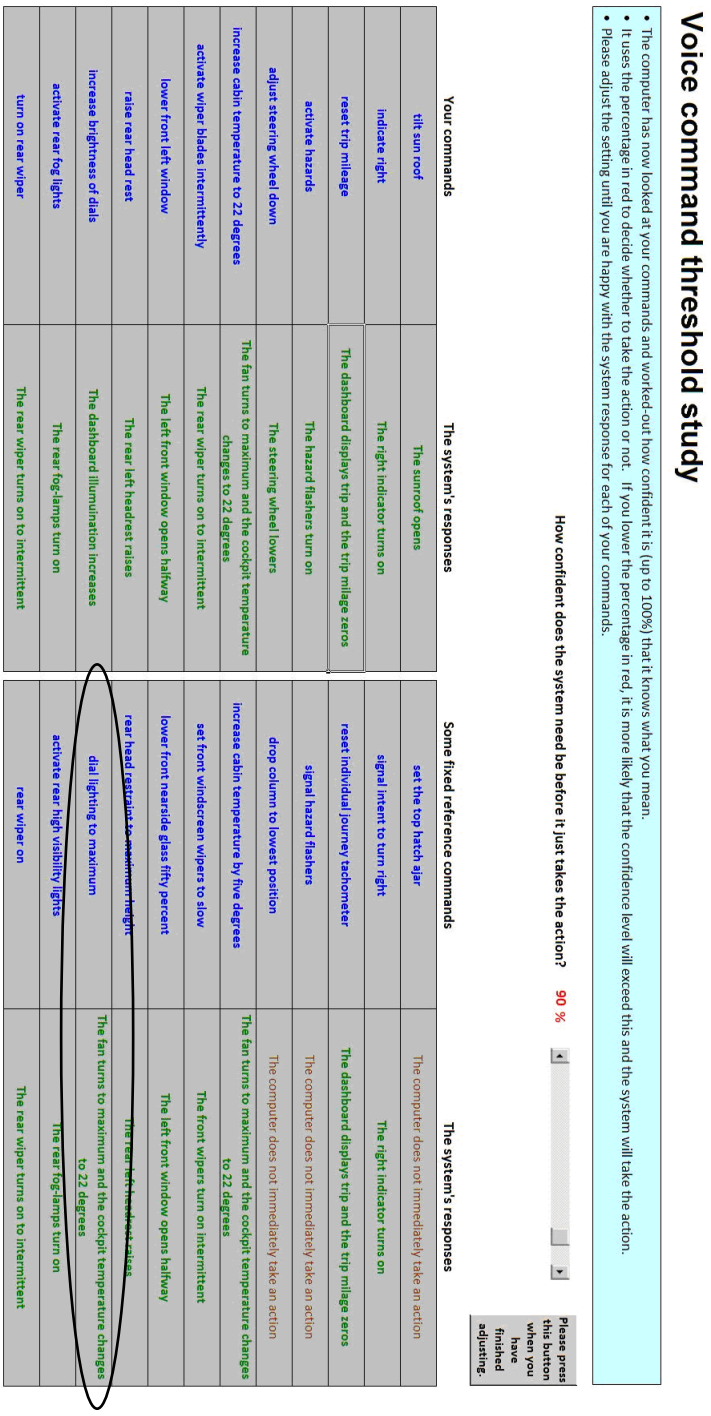



Figure 8.2. Study interface for user-tuning of 'action / confirmation' threshold (study 6).

Once participants had finished adjusting the threshold for whether to take an action or not, the next threshold was shown. The second threshold only affected commands which were not identified (by the level of the first slider) as immediately taken. Participants used the second threshold to tune the confidence level at which the command processing method would either confirm the action it considers most likely or alternatively ask for the command to be rephrased. Prompts again highlighted the effect of the adjustment available at this time. The interface for this threshold is shown in Figure 8.3.


Figure 8.3. Study interface for user-tuning of 'confirmation / rephrasing' threshold (study 6).

The final threshold adjustment allowed participants to identify the similarity in confidence at which the command processing system should consider that it cannot differentiate between the two most likely candidate actions. The interface for this threshold is shown in Figure 8.4.


Figure 8.4. Study interface for user-tuning of ambiguity sensitivity threshold (study 6).

In the last part of the threshold study, participants were presented with all three sliders and given the option to further adjust them to achieve the best settings. The settings participants had chosen during individual setting adjustment were displayed next to the three sliders as shown in Figure 8.5.

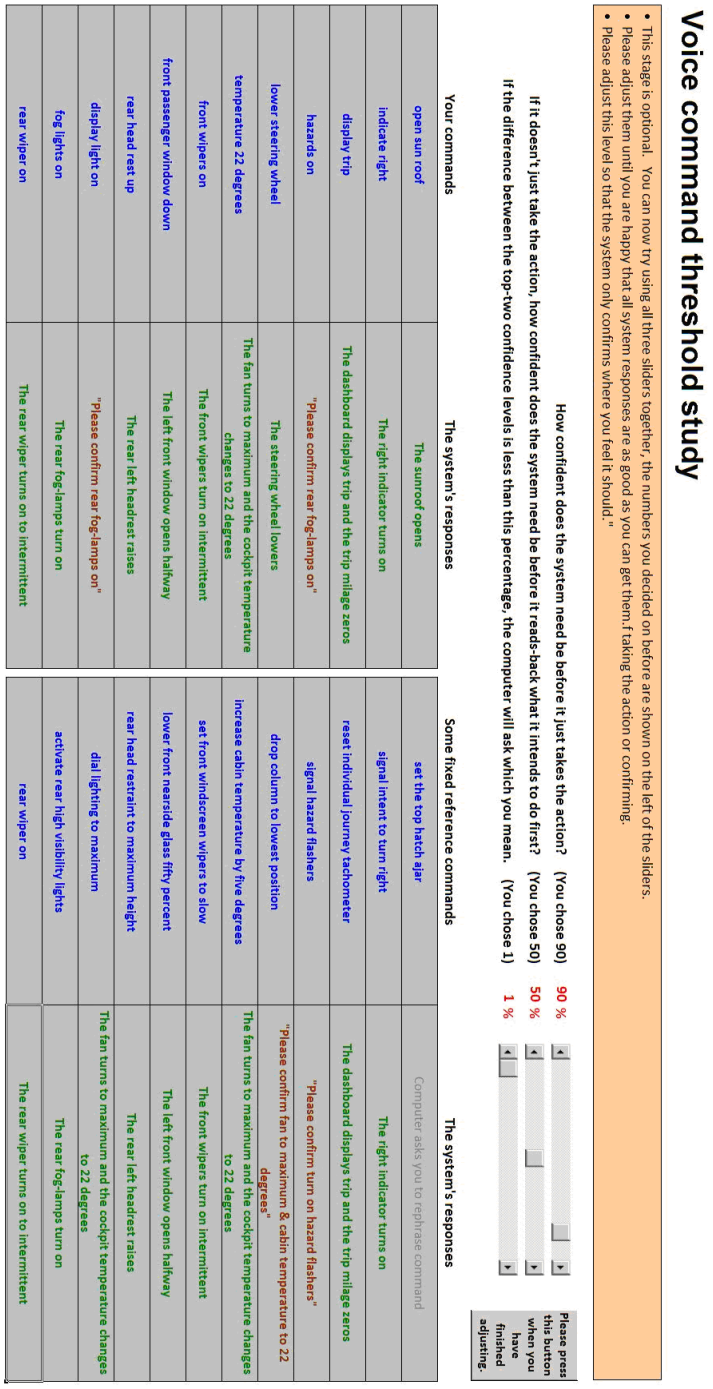



Figure 8.5. Study interface for simultaneous user-tuning of all response thresholds (study 6).

In the last sheet, participants were asked if they were happy to give their gender, indicate their age within a set of ranges and state whether English was their first language. Instructions were also given for the participant to save and return the Excel workbook to the researcher via email.

The command processing method developed in chapter 5 was used to generate responses during this study. The prompted actions and prompting animations were those used in study 3

and the typed commands given in study 3 were used as the corpus which formed the basis for calculation of confidence.

### 8.2.2 Participants

Potential participants were contacted by email asking for their assistance with a description of the purpose and content of the study. The study interface was attached to the email and recipients were invited to open this study if they elected to participate.

This survey was sent via a contact email to 500 people within Jaguar Land Rover, including both genders and ages ranging from 20 to 65. These potential participants excluded any person with knowledge of HMI methods. 55 recipients of the email elected to participate in the study and all participants elected to provide both their gender and their age range.

### 8.2.3 Thresholds

Three thresholds for calculated confidence were included in this study:

- An immediate-action threshold. Above this threshold, the command processing method would indicate to the participant that a stated action would be immediately taken.
- A (lower) confirmation threshold. Between the confirmation and the immediate action thresholds, the command processing method would ask for confirmation of the action before acting. Below the confirmation threshold, the command processing method would ask the participant to rephrase the command.
- An ambiguity threshold. If the two actions with the highest confidence differed in confidence value less than this value, the command processing method would ask the participant to choose between the two actions.

## 8.3 Results

The measures considered from this study included:

- The mean thresholds chosen by participants
- The similarity of the mean threshold chosen by participants to those estimated in study 1.
- Degree of fit of participant response to possible parametric models
- Whether participant responses are single or multimodal
- Whether threshold levels are related by individual participants
- The degree to which threshold models overlap
- Given the threshold model overlap, the degree to which optimal threshold settings may satisfy a population of users.

Participants initially adjusted individual thresholds to enable them to become accustomed to the effects of these adjustments. Results are taken from the second stage where participants were able to adjust all three thresholds together as this allowed participants to observe and control both individual and interaction effects of the three thresholds. Figure 8.6 shows a dot plot for participant settings for each of the three thresholds.

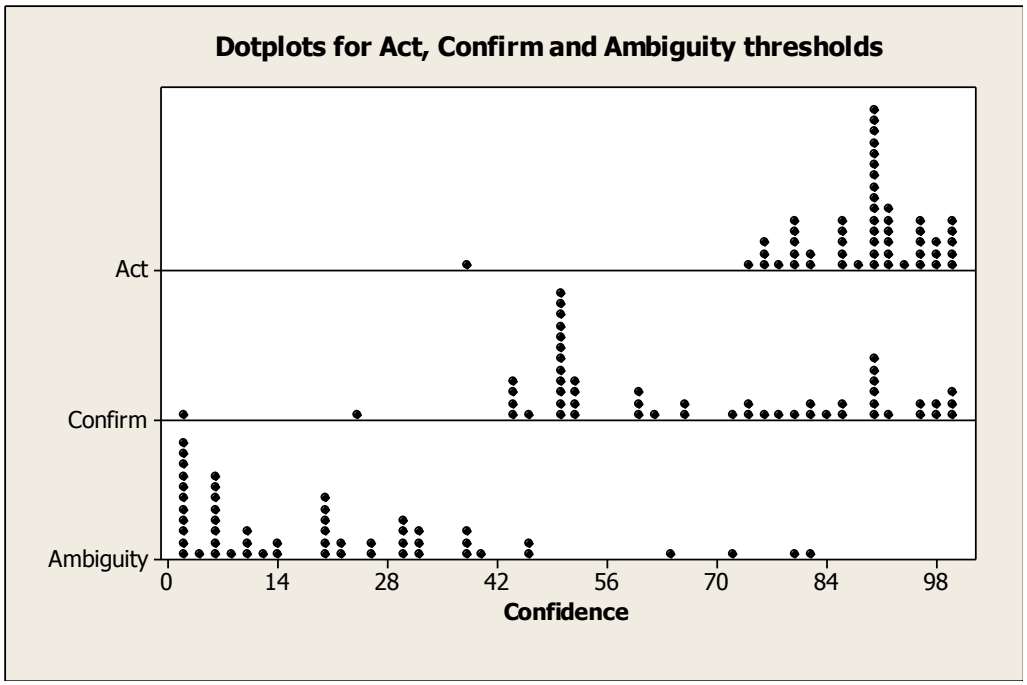



Figure 8.6. Threshold settings selected by study participants (study 6).

With the exception of one outlying sample, the set of results for the immediate-action setting suggest a parametric distribution and a possible threshold with an approximate confidence value of 70. Results for the confirm threshold do not show a clear grouping or suggest a parametric distribution and cover a wide spread of the confidence range including the area delimited by the immediate-action threshold. The ambiguity threshold shows a wide spread of preferred levels. The data appears to follow a logarithmic distribution.

The wide spread of results and significant overlap suggests that a single threshold setting may not be acceptable for a population of users. It may be possible to identify differing thresholds which better meet participants' needs if categorical subsets of users could be identified and the results subjected to rational sub grouping.

Since the threshold for confirming an action only affected those commands for which there was insufficient confidence to immediately take an action, the confirm threshold may interact with the immediate-action threshold setting. A relative confirm threshold was therefore derived as the confirm threshold value subtracted from the act threshold value. Figure 8.7 shows a frequency histogram of the relative confidence measure and Figure 8.8 shows a normality plot of the same data.

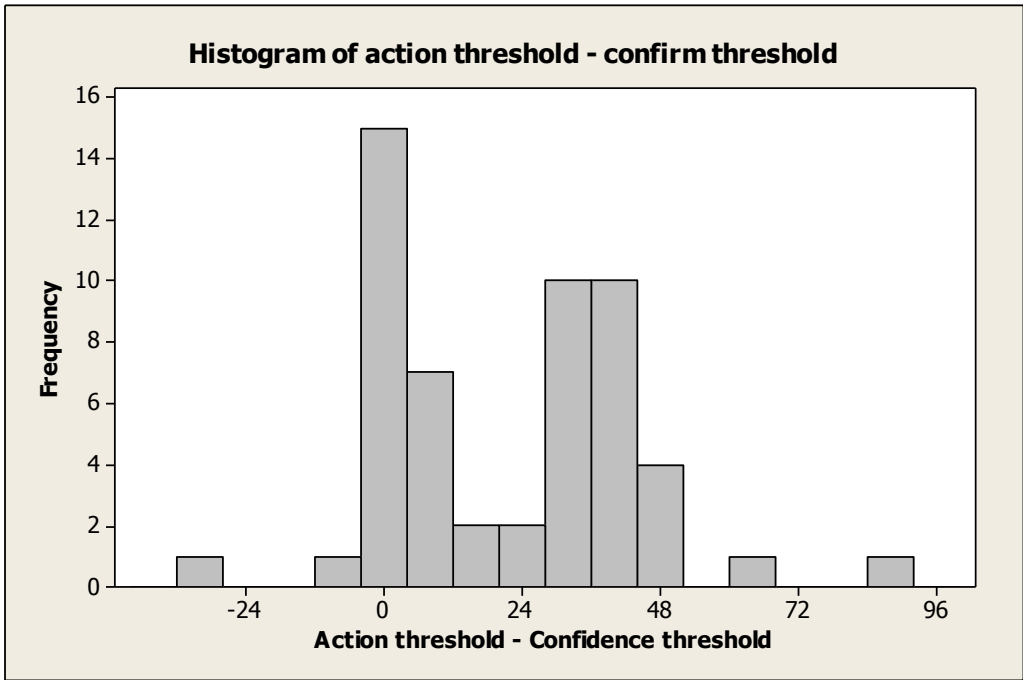



Figure 8.7. Distribution of gap between act / confirm and confirm / rephrase thresholds (study 6).

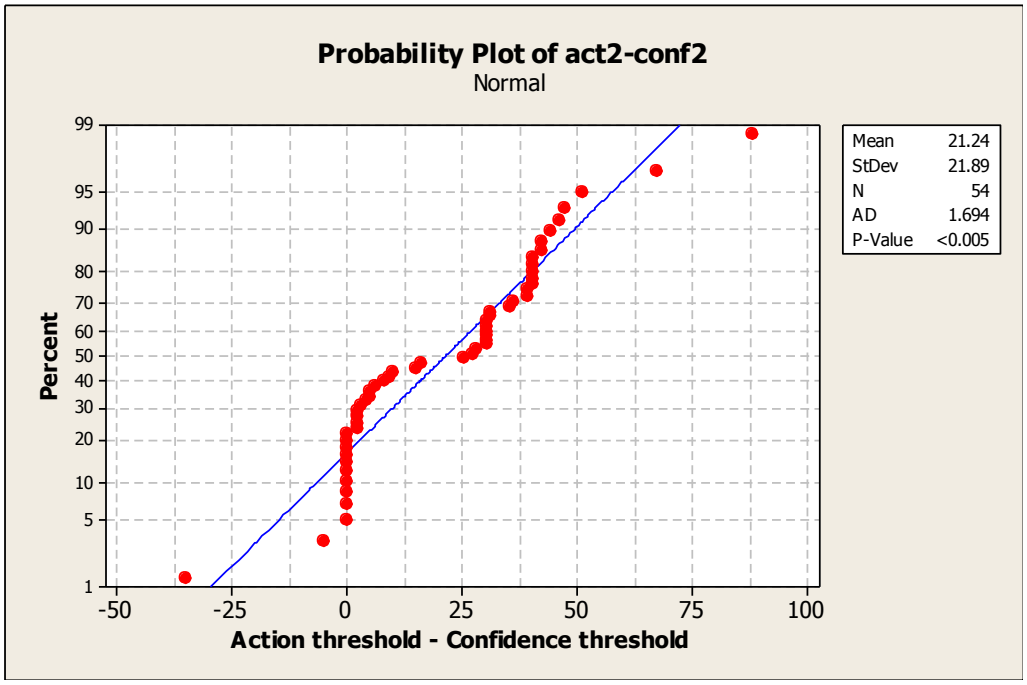



Figure 8.8. Normality plot for difference between act / confirm and confirm / rephrase thresholds (study 6).

The pattern shown on Figure 8.8 suggests that the results may be bi-modal or multi-modal. To address wide spread of data and possible multi-modality, the additional measures taken in the third part of the study were assessed for statistical significance and potential for rational sub grouping.

No significant difference was found between genders for the three threshold values or the relative confidence level. Regression tests for participant age and each of the three thresholds also showed no significant relationships. However, correlation was found between participant age and relative confidence level [$F(1,51) = 6.97$, $p = .011$] with an $R^2$ of .12 as shown on Figure 8.9. Whilst this has potential for reducing the range of potential threshold values (assuming user age could be identified), it does not explain the multi-modal nature of the data.

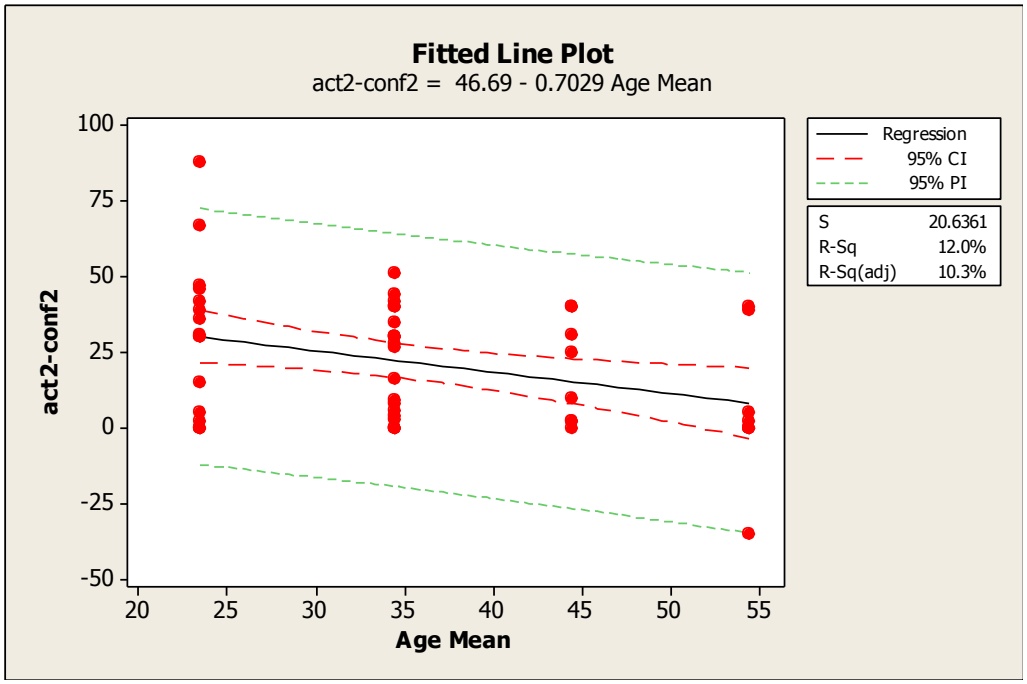



Figure 8.9. Regression of mean age and difference between act / confirm and confirm / rephrase thresholds (study 6).

The dual peak of the histogram in Figure 8.7 and the double curve in Figure 8.8 suggest that there may be a bimodal data set relating to the relative confidence setting. To assess this possibility, a dot-plot of the relative confidence threshold data was plotted as shown in Figure 8.10. This suggested a threshold for separating the bimodal data and this allowed investigation of a factor which might differentiate the data above and below this threshold.

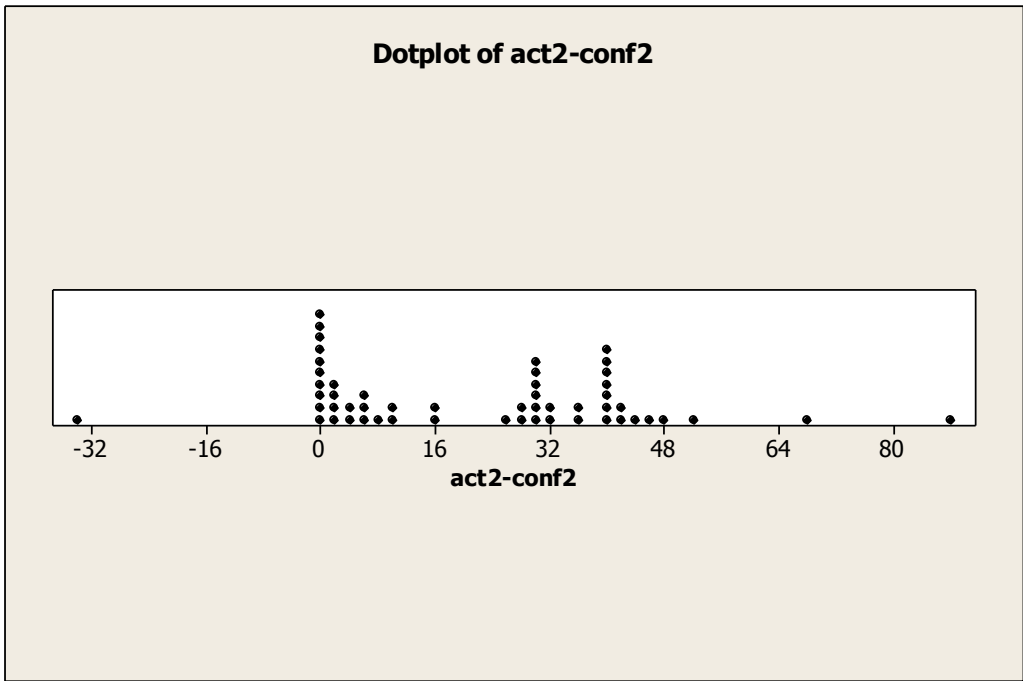



Figure 8.10. Dot-plot of difference between act / confirm and confirm / rephrase thresholds (study 6).

A dot plot in Figure 8.10 shows a possible separation in the data set occurs at a confidence level of 20. Since the measure used was relative confidence level (immediate action threshold – confirmation threshold), the act and confidence threshold values were separated into two data sets by this relative confidence level. The two sets are shown in Figure 8.11.

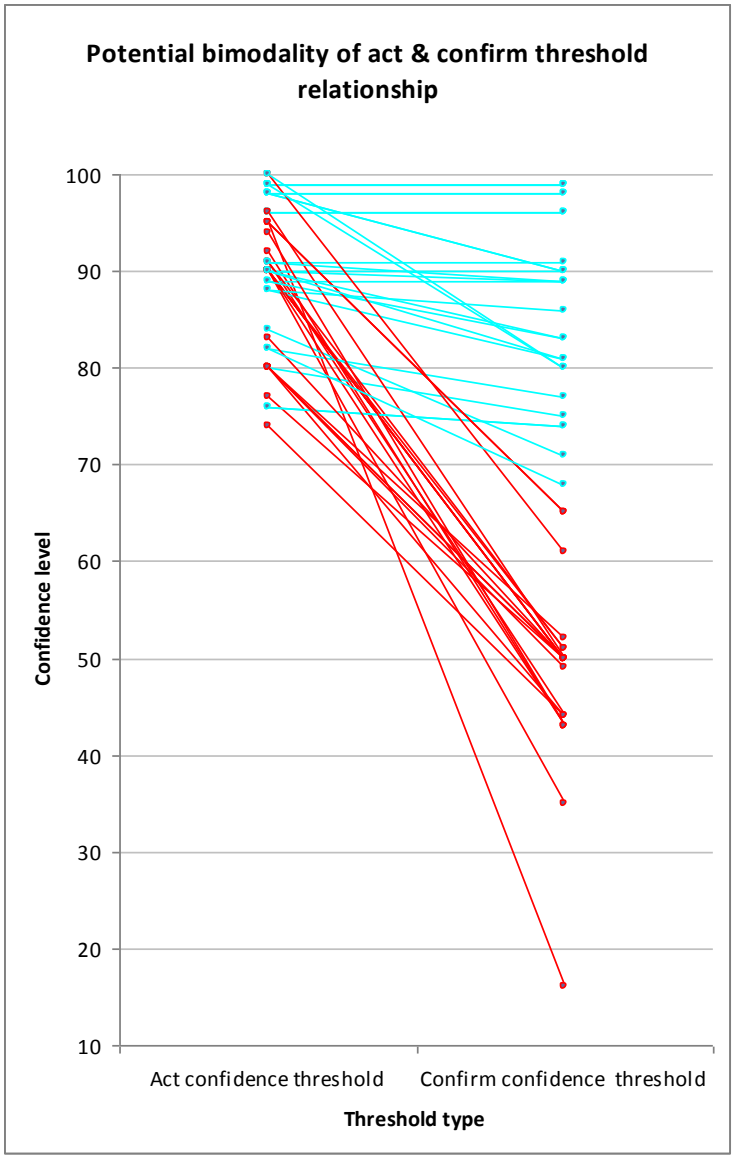



Figure 8.11. Comparison of participant responses for act / confirm and confirm / rephrase thresholds highlighting potential bimodality (study 6).

The two data sets (separated at a relative confirm threshold of 20) are separated at the confirm threshold but overlapped at the act thresholds. This suggests that any factor which causes bimodality may relate to the confirm threshold. Individual participant response sheets were investigated to contrast the two data sets and it was found that in the cases where relative confirm thresholds were below 20, the threshold setting did not allow any requests for commands to be confirmed to be displayed. In the cases where relative confirm thresholds were higher than 20, one or more confirm requests were shown to the participant.

To manage the two populations, a user of this method in an automotive context could be presented with a preference option to disallow voice interface command confirmation thus only allowing immediate action or a request to rephrase (or in the case of a lower voice recognition

confidence, repeat) a command. This would constrain the results to the subset of data for which the relative confidence threshold was above 20.

The effect of constraining the data to this subset is shown for the act threshold in figures 8.12 – 8.14 where the graph of the left shows the full data-set and on the right, the data-set for relative confidence values above 20 are shown.

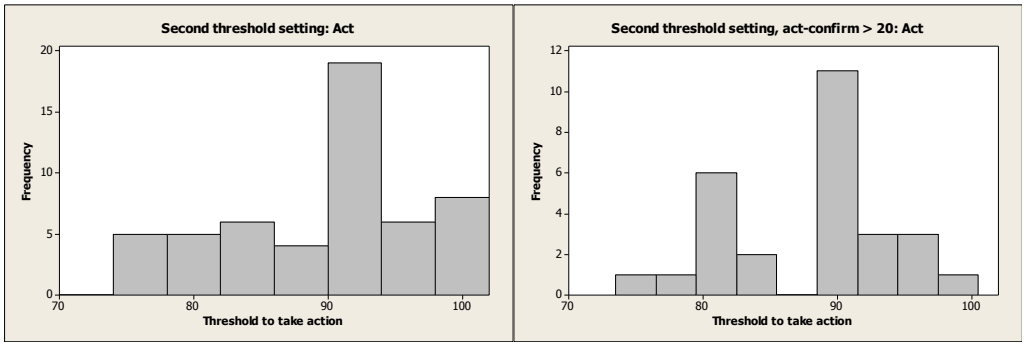



Figure 8.12. Histograms of two potential modal groups for act / confirm threshold (study 6).

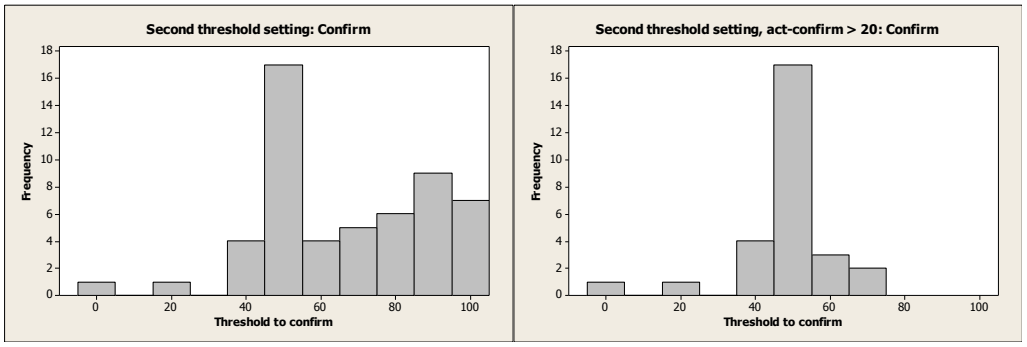



Figure 8.13. Histograms of two potential modal groups for confirm / rephrase threshold (study 6).

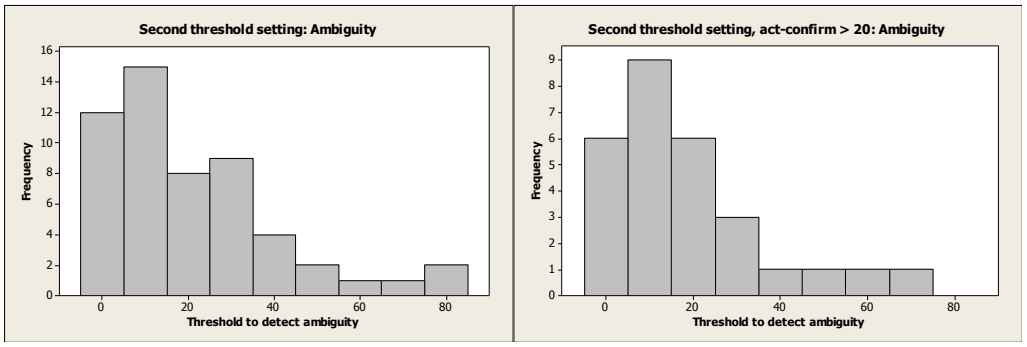



Figure 8.14. Histograms of two potential modal groups for ambiguity threshold (study 6).

The lower limit value for the immediate-action threshold is unchanged and there is minimal effect on the upper value for the disambiguation threshold. The confirm threshold is however now constrained to a lower maximum value and the data is centrally grouped.

The resulting data for the three thresholds is shown in a dot plot in Figure 8.15.

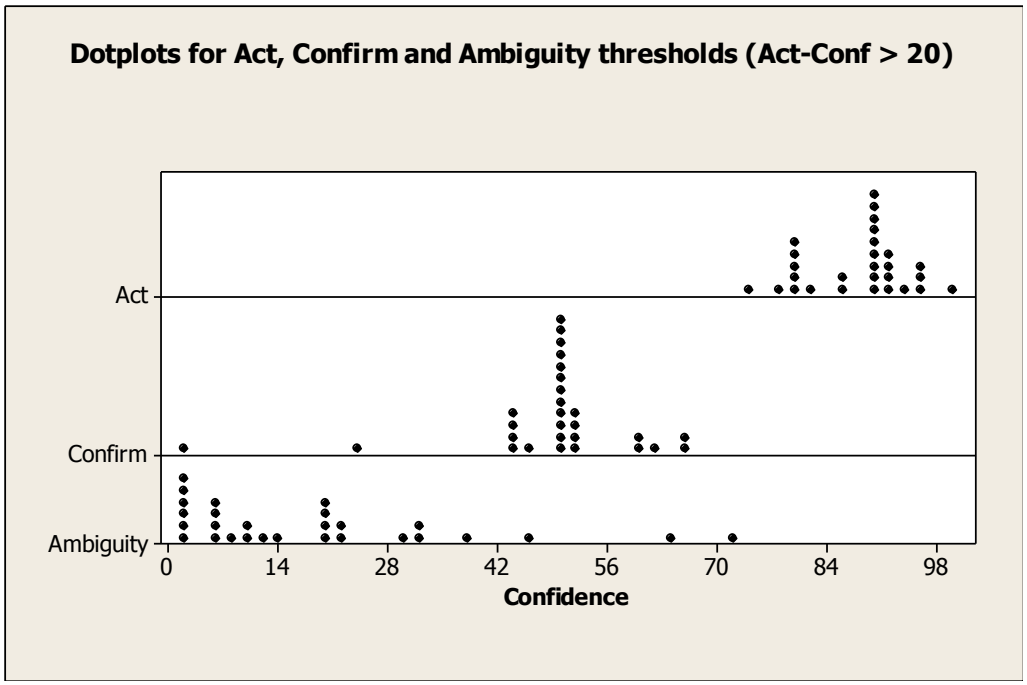



Figure 8.15. Dot plots of all threshold for one potential modal group (study 6).

## 8.4 Discussion

All of the study participants understood both the user interface of the study tool and the individually introduced response sliders. They were able to observe system phrases responding as they adjusted the sliders and to optimise all three thresholds to their preference.

Figure 8.6 showed no clear separation between the immediate-action and confirmation thresholds. It was found however that participants' had, in all cases where an overlap existed, applied threshold settings which disallowed any confirmation dialogue. It would be possible to isolate this subset of users by allowing them to disallow dialogue via a voice interface menu option and in such a case, a threshold between immediate-action and confirmation would no longer be relevant. On this basis, this data was re-plotted showing only the subset of participants who allowed a confirmation dialogue (Figure 8.15) and this showed a separation between the immediate-action and confirm thresholds. In the case where the set of system users comprised the study participants and people of whose choice of command phrasing is similar to that of the study participants, it may therefore be possible to implement this method with a single fixed threshold values which are suitable for all users.

That some participants allowed confirmation messages and some did not may be related to responses to question 4 from study 1 (chapter 3). Participants were asked 'would you rather be asked to repeat a command or be prompted to go through a command step by step if the system doesn't understand you?' Of the 19 responses, 12 stated that they would prefer or require that they be allowed to repeat a command. Whilst this is not entirely the same as a simple confirmation, it may relate to a preference as to whether a participant is prepared to

converse with the voice interface. If this does indicate that many participants do not wish to converse with a voice interface, this may have implications for future methods which could be based on human-computer dialogue. One possible reason for reluctance to enter into a human-machine dialogue may be noted from Figure 8.9 which shows that older user are less likely to accept dialogue. Possible reasons include increased sensitivity to cognitive load with age noted by Wolters et al. [74, P.7] and the potential that younger users are more accustomed to human-machine interaction.

Whilst a common threshold was possible within the participant set, human factors may require different values for participant sets from different geographical locations. Since no difficulty was observed in using a remote, crowdsourced method for this study, this may be possible without scaling resource requirements needed to obtain the data.

Participants were asked to first adjust one threshold at a time (to limit possible confusion as to individual slider function and interactions of thresholds) and were then allowed to adjust all three thresholds. Settings selected when all three sliders were available differed only marginally from the individually set levels and this suggests that the interaction effect of the thresholds did not affect users' preference for threshold levels.

The mean setting for immediate action was 87.8% and the average confidence suggested as needed for immediate action by study 1 participants was 88.3%, differing by only 0.5%. That the values are very similar suggests that participants in the survey may have had a perception of the consequences immediate-action confidence threshold that matched the actual effect observed by participants during the study described in this section.

The study tool was built into an Excel spreadsheet which showed no outward signs of the complexity of the command processing method. It might therefore have been expected that participants would not have high expectations of their commands being correctly identified. However, following completion of the survey, three of the participants returning the study form by email noted that they thought that the study tool was not working correctly as on occasion it was unable to understand their commands. Whilst this is a small sample, it suggests that user expectation of computational command-phrase processing may in some cases be unrealistically high. This may relate to the JD Powers results noted in section 1.2.

## 8.5 Conclusion

Research question 5 addresses the potential to develop a crowd-sourcing method of obtaining geographically tagged threshold levels for command interpretation system responses to optimally manage differing confidence levels when identifying an intended action. Since all study participants understood both the user interface of the study tool and the individually introduced response sliders, this supports the potential to employ crowdsourcing to elicit

response threshold preference information from users. Common optimal-response thresholds for all participants were identified using the crowdsourced data obtained in this study. This supports the use of thresholds for response selection and suggests that further work to consider whether thresholds remain constant for other dialects, languages and cultures may be of value.

It was noted that to enable fixed threshold values to provide optimal responses in all cases, it may be necessary to provide users of natural language command processing methods with the option to disallow confirmations and/or dialogue methods for disambiguation.

In chapter 9 the overall findings and implications of the studies performed in this thesis are discussed. The degrees to which the research question stated in section 1.4 have been met are considered and opportunities for future research are proposed.

# *Chapter 9: General conclusions and discussion*

## 9.1 Discussion

### Thesis background

This thesis was initiated by a commercial automotive survey which found considerable dissatisfaction with voice interfaces. Safety benefits resulting from using this control method were noted by Graham and Carter [3, p.155], Burnett and Porter [4, p.523] and Greenburg et. al. [23, unformatted] and there appeared to be an opportunity to investigate how to improve user satisfaction and therefore usage. Improvements could relate to voice interface performance, user acceptance of voice interface performance or improving the way that poor voice interface performance was managed. Since considerable research into voice interface performance was already ongoing, the initial direction of research was within the Human Factors discipline, involving consideration of the human-machine interface. Reviewing current commercial systems noted that they were fixed-command methods with limited habitability, that such methods could not distinguish between recognition errors and interpretation errors and that system responses consequently had a limited ability to guide users to improve interaction. Reviewing command processing methods under development suggested that whilst such methods could separate interpretation errors from recognition errors, they were often based on large corpuses of information (examples include Banko and Brill [75], Gale et. al. [76] and Banko and Brill [77]) which may not match voice command syntax and that collecting such comprehensive data across the geographic area in which an automotive solution might need to be deployed could be prohibitively resource intensive.

### 9.1.1 Research questions

Research questions were developed from literature reviewed and from the results of studies 1 and 2. Figure 9.1 shows the logical path by which the initial premise of user dissatisfaction with current Voice Interfaces led to the focus of investigation into potential methods to address the research questions stated in section 1.4. These comprised:

#### Research question 1: Method of gathering command sample data

To investigate the potential to develop a crowd-sourcing method for gathering geographically tagged, automotive speech interface command-phrase samples such that effect on choice of vocabulary is minimised.

#### Research question 2: Validity of command sample data

To assess the degree to which choice of vocabulary varies between a spoken modality (which would be used to operate an automotive voice interface) and a typed modality (which facilitates crowdsourcing).

### Research question 3: Method of command processing using command sample data

To assess how effectively spoken commands may be interpreted using only crowdsourced example commands, focussing on a simple token-based command processing approach.

### Research question 4: Habitability of command processing method

To evaluate the degree of success and confidence with which the command interpretation method is able to identify actions intended by automotive-domain commands and the effect on success and confidence of action identification in cases where due to the voice recognition challenges present in an automotive domain, the command phrase has not been fully recognised.

### Research question 5: Method of gathering response threshold data and effect on command processing

To investigate the potential to develop a crowd-sourcing method of obtaining geographically tagged threshold levels for command interpretation system responses to optimally manage differing confidence levels when identifying an intended action.

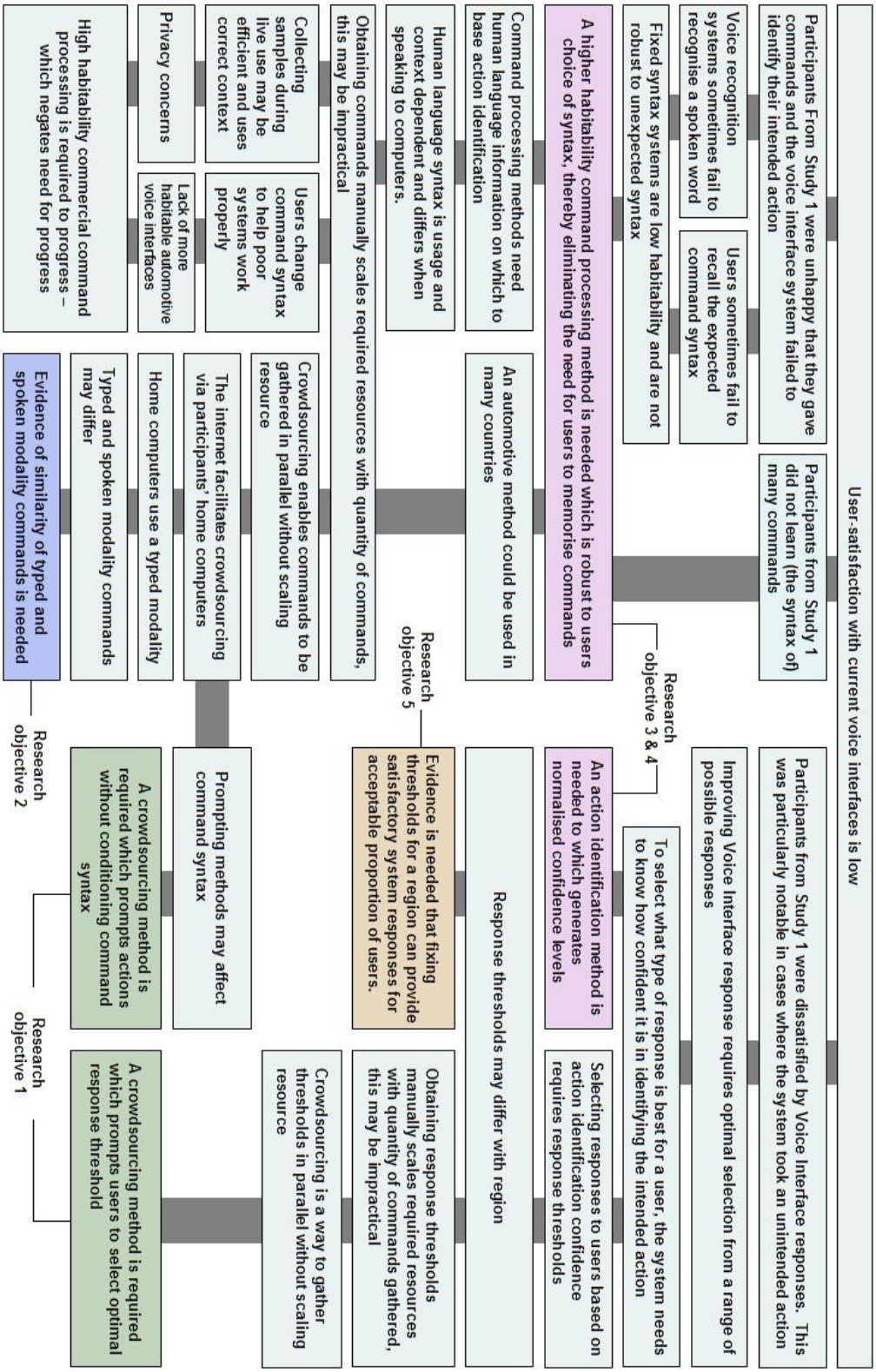



Figure 9.1. A logical deconstruction of user-satisfaction with voice interfaces, identifying the focus of research.

The requirement that a voice interface could be deployed across a broad region required consideration of what information might be obtained from across this region in an efficient, scalable way. Since the ideal context for collecting data would be live-usage sample recording and such data would comprise examples of commands given, this led to the development of

methods to obtain command sample data (research question 1) such that commands used a similar phrasing to those which might be spoken in an automotive context. One option was to use simulators to obtain this information and such equipment can vary in cost and representativeness. Jameson and Jameson [78, p.5] note the need for researchers to be confident that low cost simulator methods are sufficiently representative and also note [78, p.4] that simulators cannot entirely replicate the real world. Groeger et. al. [79] found that speed estimation can be underestimated in simulator studies. The use of simulators was not pursued due to scaling of costs to the potentially large number of samples required. A less expensive option was the use of researcher-based surveys, however whilst Letts et. al. [80, p.6] note that interviews are relatively inexpensive, the need to scale data gathering over a large area and many samples made this nonetheless impractical. Crowdsourcing was selected as the focus of research question 1 (section 1.4) as it allowed simultaneous gathering of many command samples from a wide geographical area without resource scaling.

Samples from typed (which facilitates crowdsourcing) and spoken modalities were compared and differences were noted in choice of vocabulary (research question 2) but it was not possible to identify the effects of noted differences on success of voice interface action identification. The development of a command processing method (research question 3) was not initially planned as part of this thesis; however it later became clear that such a method was needed to assess the effects of corpus modality on voice interface habitability. The developed method facilitated assessment of the habitability of a (simple pattern-matching) voice interface system for a specific set of command samples (research question 4) and allowed the consequence of gathering command samples using a typed modality to be assessed. Development of the method took several iterations and considerable effort; the end result was a pattern-matching method which proved surprisingly effective within the scope of the language and dialect studied. There was a need to ensure this system was robust to voice recognition failures (research question 4) (word omissions were used for this evaluation) and once this had been completed, the method was then sufficiently developed to allow research to progress to system responses (research question 5). For the final system response study, the command processing method was too effective and the study was adjusted to force participants to choose a preferred compromise between different responses. For the set of participants, the results showed that all participants could be satisfied by common system responses based on user-optimised thresholds.

Before the methods described could be developed into a full voice interface, further research would be needed to consider different cultures, languages and dialects. Assessment within a live automotive usage situation would be also be needed. In addition, other methods are required before a voice interface can be implemented. Examples include methods to identify a new user's language and dialect, identifying when a different person is driving and evaluating what response syntax is most acceptable to users. However, within the scope of this thesis, the

developed methods have proven effective in addressing the commercial survey findings and research performed.

### 9.1.2 Initial investigation

Study 1 (chapter 3) describes a semi-structure interview which investigated the factors driving low user-satisfaction with current commercial voice interface systems. The set of questions was cascaded from the steps involved in operating a voice interface; additional questions considered the opinions of participants on what might improve their experience. Factors which generated the lowest levels of satisfaction and which most related to desired improvements fell within the category of low habitability (a requirement to use specific commands and a poor fit of fixed commands to participants' natural choice) and the responses of the voice interface when a command was not understood.

Current commercial voice command interface systems assessed as part of this thesis employed a fixed-command method with low habitability and therefore required users to memorise a potentially numerous set of commands. Participants in study 1 who had used such systems had however memorised only a small subset of voice interface commands. Limitations in willingness and/or ability of users to memorise commands show that fixed-command voice interface methods do not provide a scaleable solution and this raises the question of why system designers took this approach. One possible explanation is that this was the only method available when the interface was designed and that whilst limitations were understood, the expectation was that the voice interface would provide a secondary modality which users could employ to a limited extent. An alternative reason may be that system designers did not seek to fully understand user variability and underestimated the variety of possible command phrases or overestimated the proportion of users represented by the fixed command in the manner noted by Gould [39, p.97]. It was logical that a highly habitable command processing method which allows users to express commands using their choice of syntax would resolve issues relating to user memorisation of commands.

User preference for natural language interfaces was highlighted by question 11 of the semi-structured interview described in study 1. All participants stated that they would take the time to programme the systems they used with their own choice of words if this option were available. The unanimous result suggests that the preset command phrase was poorly chosen (as it did not satisfy any of the study participants) or simply that people vary in preferred command phrases to such a degree that a system which allocates one acceptable command (or a small number of variants) to each action is unlikely to fully satisfy any user.

To optimise command processing system responses (research question 5), it was necessary to consider how accurately a command processing method can identify its confidence in selecting an intended action. For example, if the method has an 80% confidence that a particular action is intended but it calculates a 90% confidence, it may take the action whereas at 80%, it should first confirm this with the user. Participants from study 1 were particularly averse to incorrect commands being enacted (and set the threshold for enacting a command at a high level),

neither did they like system-interaction dialogue (in some cases, no dialogue was preferred). In circumstances where the system is unsure of its confidence that it has identified the intended action, it must assume it is less confident to ensure that incorrect actions are not taken. This lower confidence will result in more commands being referred to the user for confirmation and thereby reduce satisfaction. The initial investigation noted the need for a natural language method which would accurately calculate the confidence with which it knew the intent of a command and would therefore accurately handle responses to users.

### 9.1.3 Literature review and assessment of current command processing capabilities

Material researched for this thesis included methods to recognise spoken phrases, methods to identify the intent of a spoken phrase and considerations of the benefits and consequences of voice interfaces. Material relating to the safety benefits of effective voice interfaces and other human factors considerations were all relevant and pertinent to this thesis. Whilst concerns over cognitive load were raise in circumstances where users needed to memorise the (unseen) context of a system using cascaded menus, the common view of research literature was that the benefits of automotive voice interface use will be considerable where such interfaces are accepted and used. Research on data-gathering methods and specifically the potential to use crowdsourced command samples as the basis for command processing was included. The greater part of user dissatisfaction with current voice interfaces related to fixed interface, low habitability systems. The concept of habitability is reviewed in section 2.5.

The process of research and collating literature on the methods for interpreting commands was challenging since much of the natural language processing literature differed from the constrained command/response methods under consideration. For example, in the case of a human-machine interface which can interact in a human-human equivalent manner (i.e. with artificial intelligence qualities), a broad understanding of semantic meaning may be needed along with the ability to contextualise a statement in terms of a world-view. Neither of these may be required for a command interface with a limited scope. The scope of this thesis is an automotive command/response system which is syntactically habitable, i.e. which needs only to identify which of a fixed range of available actions is intended by the set of commands which might be spoken in the context of driving a vehicle. For example, an AI system might be expected to understand a command ‘state how long you have been awake’. Whilst this is semantically valid and would be understood by a human, it is not relevant in the context of an automotive command system. The statement ‘allow air and light into the vehicle by means of the roof hatch’ is both semantically valid and may indicate one of the available actions within an automotive environment. However, if this wording were sufficiently obscure that no user would employ this command phrase at any time, it need not be correctly interpreted by a command/response system.

Research into the ongoing development of natural language methods highlighted factors which informed research goals from this point.

- Whilst command processing via key-word identification is possible, G. W. Furnas et. al. [21] shows that random pairs of people use the same word for an object only 10-20 percent of the time. They point out that capturing a range of words for each noun is one option to improve this result. As noted by Gould [39, p.97], this would need to involve studies with potential users. Key word systems may not correctly categorise commands if a key word is incorrectly recognised (for example where ambient noise is high). Also, where alternate noun lists grow, the potential for ambiguity must logically increase.
- More complex command processing methods may require detailed probabilistic models or grammars and these may need to be based on machine-processing of very large amounts of human generated information. Banko and Brill [75, unformatted] show disambiguation performance increasing with training material size up to and beyond 700,000 words.
- Basing command processing on existing human/human communication (for example written literature) may not provide an effective analogue for human/machine communication. Bradford [31, p.63] notes that "Human/computer communication generally consists of one syntactically complete utterance (roughly corresponding to a sentence) from the user followed by some kind of response from the computer. Human/human communication is much more varied."

These three points suggested that key word methods are not a viable solution and that basing automotive command processing on material from a different context to automotive human-machine interaction may be ineffective. Since it was possible that a large amount of human-machine interaction material may be required for command processing to be effective, this raised the consideration of how this might be obtained.

Examples of human-machine communication for an automotive command processing method could be required from across a wide geographical areas (including many languages and within these, many dialects) and this further increased the quantity of material needed and exacerbated resources required to obtain it. One option to address this was to consider what material might be gathered using a crowdsourcing method and what command processing methods might be able to make use of data which might be obtained in this way.

### 9.1.4 Sample gathering methods (research question 1)

Initially an attempt was made to devise a way of obtaining command samples using a spoken modality and a crowd-sourcing method. This comprised an application specifically designed to be sent to study participants to be completed on their own computers. The application allowed participants to give commands in response to a set of visual prompts. It employed a method by which participants' spoken words would be recorded and stored within the application. The study tool was successful in that it worked effectively on the researcher's own computers. However, on a range of other computers subsequently tested (in preparation for a full-scale study), a range of problems was noted. Some computers had no microphone fitted. Others used virus detection tools which blocked an attempt to activate the microphone. Some computers displayed prominent and distracting warnings before allowing code to run. In addition to these difficulties, there were ethical considerations in that spoken sound samples were to a degree personally identifiable. After considerable reworking and retesting, the study was set aside.

One observation resulting from this work was that a voice input modality is not in common use in home computing. The potential to obtain spoken samples could be revisited should home computing speech interfaces become common. In doing so, human response to voice recording may need to be considered. Speer & Hutchby [81, p.316] summarise a range of material which suggests that the presence of speech recording devices may 'inhibit the researcher's participation in many situations'. Mobile telephones or tablets could be used to overcome the availability of microphones, however the issues of security software remains and in addition, spoken commands would need to be transcribed for use in a corpus which would always risk incurring voice recognition errors; Juang & Rabiner [82, p.16] note that as opposed to elimination of errors, an objective of recogniser design is to 'achieve the least recognition errors'. Alternatively, human transcription could be employed, however this does not meet the criteria of scalability (i.e. resource demands would increase with number of samples required).

The potential to record natural language commands during live use required the availability of automotive natural language voice interfaces which were of sufficient capability that drivers could expect commands to be understood. No such systems were identified during research and therefore this was not possible. Constraints including scalability, context and resource/cost resulted in elimination of the options to use researchers and/or use of dedicated test facilities such a vehicle simulators to gather the information.

Research addressing question 1 focussed on how typed-modality command samples could be prompted and recorded in a way which supported a crowdsourcing data-gathering process. It was possible that command modality could affect phrasing of commands. Since crowdsourcing and a typed modality were the selected scope, research question 2 was

identified to allow assessment of the degree to which typed and spoken modality command phrases differed.

Crowdsourcing via the internet and the use of a typed modality were ways to overcome the scaling / resource constraint and the investigation into whether samples obtained in this way would result in effective command interpretation led to the need to develop a simple test command processing method.

### 9.1.5 Sample quality effect of using a typed modality (research question 2)

An initial study was performed to consider differences in command phrasing between spoken and typed modalities. The context of the study was selected as operation of systems in general use, avoiding systems commonly fitted with voice interfaces (therefore avoiding preconditioning and division between people who use voice interfaces and those who do not). Zoltan-Ford [18, p.529-530] reviewed a range of material which found that people tend to alter their style of speech according to what they have recently heard. Non textual prompting was therefore used to prevent participant conditioning. Options considered included using pictograms to indicate desired intent or using a known conventional interface and asking for the voice-command equivalent. The latter method was selected as pictograms could be variably interpreted whereas conventional controls may, in selected cases, be more consistently understood.

Whilst results from first command modality study highlighted a range of specific differences between typed and spoken modalities, the degree of commonality of command words suggested that typed modality samples may provide an effective command processor corpus and indicated that further research would be worthwhile.

The study identified a number of factors which could be improved upon in future studies. One opportunity involved the abbreviated nature of commands issued, described in a range of material summarised by Amalberti et. al. [9, p.4] which show verbal modifications when people speak to computers. In some cases, commands omitted a verb or noun, using the context to imply the missing word. It was not clear from the results whether the abbreviations would be similarly used in an automotive context. Since syntax was simplified, this may have reduced the sensitivity of the comparison between typed and spoken modalities. Two options for future improvement were the selection of commands which require a longer command phrase complexity to unambiguously state and the selection of commands from within an automotive context. A second factor which differentiated results was the subject of the commands given. Since the prompting method indicated a conventional control and asked for the voice-command alternative, some participants gave a command which replaced the conventional control and others instructed the conventional control to operate.

A second study into typed and spoken modalities was performed, incorporating the following changes:

- Focus was changed to automotive systems. This ensured that results were relevant to the intended context.
- Increased complexity actions were selected to engender more verbose commands.
- The original fixed prompting method was replaced by animations depicting desired actions taking place. This ensured that the intended subject was addressed by all participants.

In study 3, no words were used in the animations of actions taking place; this prevented conditioning of subsequent commands. There may however be some automotive actions where omitting words from prompts may be difficult. For example, if the action intended is the selection and playing of an MP3-based music play-list from a USB stick, it may be difficult to indicate the concept of a play-list without using words. The effect of using words in such cases is an opportunity for further research.

### 9.1.6 Sample quantity and quality (research question 4)

The revised methods used in study 3 achieved the intended improvements over study 2 (more verbose commands with common intent) and as such the results were suitable for a more detailed examination of modality effect on command phrases. Results showed that whilst individual participants had limited commonality of words between their spoken and typed commands, as quantity of samples was increase, commonality of words between the set of spoken samples and the set of typed samples became increasingly common. The number of samples required to attain a degree of commonality was shown to be related to average number of command words for a particular action.

The results of study 3 were of-necessity ambiguous in terms of identifying whether typed samples were suitable for use in identifying actions intended by spoken samples. Differences in command phrases were observed but without a command processing test method, it was not possible to quantify the effect of these differences. Furthermore, it was not possible to progress to investigating methods to optimise responses to users without a command processing method to facilitate this.

Research progressed to development of a simple command processing test method which could make use of the data obtained in the two completed modality studies.

### 9.1.7 Command processing test method development (research question 3)

The initial concept for the natural language test method was based on observed similarity in the samples obtained in study 2 (section 3.1). It was noted that although people used a variety of command phrases to indicate an action, this appeared to often comprise a reordering of a common set of words. A simple key-word method (which looks for common command words to process commands) was discarded for several reasons. Firstly, research described in chapter 2 noted the limitations of key-word methods. Secondly, key-word systems are dependent on the key words being understood by the voice recognition system and such systems are therefore not robust to voice recognition errors. Finally, potentially useful information would be discarded along with non-key words. For example, 'Turn the radio on and select channel one' may be a reasonable command with the noun and verb being 'radio on' and 'select channel-one' However, the word 'turn' may also be useful in differentiation command words between actions as it seems unlikely that the a driver might instruct the sunroof to 'turn' to an open condition. Consequently it was decided to use all words and initially, all sub-sentence patterns of words to pattern-match commands to the set of sample commands. One of the intents of this development process was to evaluate this method.

At the outset, it seemed a relatively simple task to compare all sub-sentence fragments of a command to all fragments of previously recorded samples and identify intended action by count of matches. However, development of the command processing method was considerably more exacting than envisaged, in part due to the need for a confidence level which was normalised across actions. The scope of the set of command processing method objectives and that they were identified as development proceeded and only summarised retrospectively contributed to the size of the task.

Objectives for the command processing method being developed as a test facility included:

1. Selects from a fixed set of possible actions.
2. Only interprets commands which intend one of the fixed set of possible actions.
3. Only interprets commands which might be spoken by users (excluding for example archaic or obscure phrasing)
4. Uses a set of command samples to identify which action is intended by a live command.
5. Requires no additional reference material which is unique to languages and/or dialects (e.g. verb-noun lists, word-sequence probability patterns etc).
6. Makes expansive use of information in samples (optimal sensitivity within data constraints)
7. Calculates a normalised (and therefore comparable) confidence value for identification of intended action

8. Robust to action types & command lengths.
9. Calculates a normalised confidence value for ambiguity
10. Interprets commands which include data (e.g. direction or degree of action required)
11. Interacts with voice recognition methods to generate an overall confidence level.
12. Learns and self-optimises during user-operation.
13. Able to provide some degree of action identification with damaged input (missing or incorrectly substituted words)

When the first variant of the command processing method was being constructed, objectives 1-5 were clear. The resulting model used a binary deconstruction of both a command and the command samples and compared the number of matches. The data-set was limited to three actions with samples devised by the researcher to allow the method of sub-sentence pattern-matching to be assessed prior to more detailed development.

Results were better than expected; all actions were identified by all commands from the corpus. However, quantity of incorrect matches was not taken into account and it was observed that the ratio of intended action to unintended action may provide this. Objectives 6 and 7 were added and the method was further developed using the data-set resulting from study 2. The data-set provided typed and spoken samples, allowing modalities to be compared.

The second stage of command processing method development calculated a probability that a command might intend each available action. This was performed by taking each action in turn and calculating a ratio of pattern matches for that action to pattern matches for the highest scoring of the other actions. This comprised a signal to noise score for each action. The intent was to generate a normalised metric and to make use of the incorrect match data in identifying the intended action. This method of calculating signal to noise was not successful as divide-by-zero errors occurred in many cases, preventing full analysis of the results. In addition, it was noted that not all incorrect match data was used and therefore the method did not make full use of the available information. It was however found that a typed modality corpus and a spoken modality corpus did not differ greatly in their ability to identify the action intended by a spoken command.

To mitigate the issue of divide-by-zero errors issues, version 3 of the command processing model calculated the signal to noise ratio for all samples for each action. Since the probability of obtaining a zero was considerably reduced, the occurrence of divide-by-zero errors was commensurately less. Nevertheless, there remained some occurrences and it was also noted that only one of the non-intended actions was used in the calculation. The results of model testing showed that the set of pattern-matches for non-intended actions varied considerably and there was an opportunity to use this as a signature for an intended action.

The fourth version of the command processing method operated in two stages. In the first stage, a recursive method was used to create a model for the expected degree of pattern matching for each action, should a specific action be the one intended. For example, if action 1 were intended, each of the actions might be expected to generate a level of pattern matching, including both action 1 (the action intended) and all other (non intended) actions. The fit of pattern-match results to this pre-established set of expected responses was then used to establish the probability that each action might be the one intended. This provided a method which was able to calculate a normalised measure for the actions intended by commands and used all of the available data.

Pattern matches to the set of models varied in a random fashion such that as number of actions increased, the probability of obtaining a higher degree of matching by chance increased. Again using a recursive method to assess this effect, it was noted that degree of unintended match differed in a logarithmic pattern and it was noted that this might provide a way to eliminate this effect. The fit of the most likely action against a logarithmic model of the other actions was therefore used during the second stage as a measure of confidence. This established a model of the less likely actions and identified the likelihood that the most probable action was not a part of this set of non-intended actions (i.e. assessing as a deleted residual). It was noted that this could be repeated for the second most likely action for the purpose of checking for command ambiguity, which met the requirements of objective 9.

Version 4 of the command processing method was developed to interpret textual data. To consider objective 10, some of the actions assessed required data to fully enact; these were interpreted with the data remaining as part of the command. For example one prompt showed the front left-hand window being opened halfway.

Voice recognition methods identify the degree to which spoken sounds match expected speech models. They may then derive a value of confidence for each possible word. One method of using this data would be to take all optional words above a confidence threshold and try all combinations using the command processing method described. This would then provide an overall confidence level as noted in objective 11. Whilst this was not attempted as it was outside the scope of this thesis, there is no impediment to the method developed in chapter 5 being used in this way.

The potential for self optimisation to which objective 12 refers was noted as present in all pattern matching methods. In cases where a voice command is enacted and is not thereafter repealed by the user, the command given could be added to the corpus of commands thus improving command processing method.

The need for objective 13, the ability to operate despite damaged input was noted when considering key-word systems. In the case of such systems, if the key word cannot be recognised, the command may not be identified despite the remainder of the command containing sufficient information for a human to identify the intent. Whilst it was desirable that the command processing method should be robust to unrecognised words, it was not required for assessing modality effects on action identification or response thresholds but was important in the context of the set of proposed methods. It was therefore assessed as part of chapter 7.

The method developed was not intended to be commercially viable but to be sufficiently sensitive that it could establish the viability of using a typed modality corpus for a pattern matching command processing method and also to allow response thresholds to be studied. The method resulting from the development process correctly identified intended actions in nearly 90% of cases and was more effective than was anticipated. Whilst this was very close to the level of command identification required by the participants of study 1, whether this would be good enough to significantly increase voice interface usage was not easy to identify. One possible measure for whether a method might increase voice interface usage is if it were able to facilitate users giving command samples which were more suitable to form a corpus of example commands than those used by the system to elicit the commands. In this case, when a system was commercially released, commands issued during live usage could be recorded and included as part of the command processor corpus. This would then result in self-sustaining improvement as the set of command samples is optimised over successive usage. Further research would be needed to establish whether the method devised in chapter 5 achieved this effect.

The completed model was used to assess the results from study 3 and concluded that crowdsourcing command samples using a typed modality does not have a statistically significant effect on identification of actions. It also identified that some actions are inherently ambiguous and this suggests that methods to manage expected ambiguity may be needed; increasing sample size may be one option to manage this factor.

Results from study 3 were again assessed using the completed model, omitting words to assess the robustness of the method to commands only partly recognised by the voice recognition system and also the interaction between damaged commands and corpus modality. All corpus commands were assessed several times, omitting a different word in turn, resulting in a large number of tests. The results showed no discernable difference between the performance of the typed and spoken modality corpuses and the degree of pattern-matching was not degraded due to the omitted command words. That the method made use of all words in a command to generate models for expected pattern matches was intended to provide a degree of robustness against a single word being misunderstood. If only nouns and verbs were useful when pattern-

matching commands to command samples, commands omitting these words would not be recognised. Since the proportion of commands for which the intended action was correctly recognised was unaffected by word removals (including verbs and nouns), this suggests that pattern matching all command words assisted in identification of intended actions and improved robustness to word omissions. One option for future development would be to further investigate this performance.

### 9.1.8 Optimisation of the Human-Machine interface (research question 5)

The command processing method developed in chapter 5 allowed human-machine interaction with a natural language voice-command interface to be studied, including optimisation of system responses. In cases where a command is presented to a voice interface such that the subject, action and all required data are present, the only response required would be for it to take the action and possibly confirm this to the driver. Responses from the voice interface to the human user may therefore relate primarily to managing situations where the voice interface has insufficient confidence to enact a command. Shneiderman [83, p.74] notes as part of eight 'golden rules' for interface design, the need for interfaces to offer informative feedback.

Methods available for managing confidence (in understanding the action desired) below that required for an action to be taken include:

- Asking if the action determined by the system as most likely is the one intended
- Asking if one of the two most likely actions is the one intended
- Asking for a command to be repeated
- Asking for a command to be rephrased

Study 5 considered the degree to which confidence thresholds could be identified which would select system responses which satisfied all users. The study was performed using a crowdsourcing approach to assess the potential to use this method for such studies and so that a larger number of study participant responses could be obtained. Participants gave commands for a prompted set of actions and then adjusted response thresholds to optimise responses. The study was designed such that participants needed to choose between compromises, for example allowing some commands to result in unintended actions or to prevent this, requiring confirmation requests for some correctly identified actions. One observation during development of this study method was that the command processing system was too effective. The actual confidence-level calculated by the command processing method was artificially degraded to generate an increased number of 'error' management situations.

Pre-study trials showed that the action of the threshold adjustment sliders was not easily understood by study participants. Introduction of each slider individually was found to help and descriptions of the effects of each control were adjusted until clear. Once all three sliders

were introduced, participants were allowed to adjust all three together. Results showed that there was not a single confidence threshold that would satisfy all participants. However, participant responses showed evidence of bimodality and investigation of the settings used by participants revealed that one of the two populations had set thresholds to prevent confirmations from being shown. Since this could be achieved by a set-up option in the voice interface menu, this data-set was eliminated and it was then possible to identify common response thresholds for act / confirm, confirm / rephrase and for detecting ambiguity. One opportunity for a future study would be to provide a selection to disable interface features such as action confirmation to consider more closely the observed multimodality.

It is possible that the common thresholds shown in this study may not be optimal in different geographic regions and investigating this may provide an opportunity for further study. As study 6 was performed using a crowdsourcing method, further research could be simplified by using this approach.

Two of the study participants provided similar anecdotal comments following the study. These participants both suggested that the study was flawed as it had failed to correctly enact all of their commands. It would seem that these participants expected a command processing method to work flawlessly (as the method may have done without applying an artificial bias to reduce confidence level to ensure that candidates needed to select between different non-ideal outcomes). This was surprising given a fairly ubiquitous perception that voice interfaces have a limited capability and one possibility is that users consider natural-language methods to be sufficiently different to existing fixed command methods that they set-aside their previously limited expectations. Since high confidence in a voice interface is needed to elicit commands which will form an optimal corpus, it may be worthwhile to encourage this view during data gathering and to use marketing methods to encourage this view during commercial release.

### 9.1.9 Crowdsourcing command samples for automotive voice interfaces

Command interpreters require information on which to base a decision as to which action is intended. In cases where there is a need to interpret commands from a range of languages and dialects, this may amount to a great deal of information.

In this thesis, several methods of gathering information were considered including surveys and vehicle simulators. Crowdsourcing became the focus of study because it offered the opportunity to collect information without needing resources which scale with quantity of samples and the geographical area over which samples were to be collected.

When designing a crowdsourcing method, one challenge is the communicating of an intended action to study participants without conditioning their choice of command phrase. The methods demonstrated in study three used an animation of the intended action being carried

out. This eliminated text prompting and prevented conditioning of study participants' choice of command phrase. However, it is not possible to provide animations for non-visual changes; for example activating heated seats. Also, in the case of display screen responses, these often show words or numbers which are required to communicate the intended action. In the case of automotive applications, non-visual responses are few in number and in the case of display screen commands such as 'select play-list summer hits', visible words could be restricted to the selection (in this case 'summer hits') and responses could be gathered using a variety of names and numerical values to average-out any conditioning effect. This method of prompting may not be so effective in other, non-automotive contexts where responses are largely non-visual.

The animation method assessed in study 3 was demonstrated to elicit commands which address the intended action without the need for potentially conditioning text and this method can therefore be used effectively for gathering automotive command samples for majority of actions. Further research on creative approaches to minimise the potential to condition study participants where there is no visual change may be an opportunity for further research.

A further challenge is the difference in the context of issuing commands during a crowdsourcing survey and issuing commands while driving a vehicle. Studies performed in this thesis do not consider this effect and this is an area for further research. One option may be to look at how to replicate driving influences during a crowdsourcing survey, for example introducing a time constraint or a secondary cognitive workload to better simulate driving context.

For a pattern matching command processor, it is also important that the person issuing a command expects the command to be properly processed. If this were not the case then as user-confidence increases and assuming that this may affect choice of command phrase, the capability of such a system would gradually reduce as the command samples and live commands become dissimilar and the system would effectively be self-limiting. In the case of the survey performed in study three, the command was given following observation of an action being taken and therefore the command was already predicated on a successful outcome. In addition, unsought comments following the final study asked why the interpreter was unable to understand some commands indicate anecdotally that study participants assumed a high degree of system capability.

The opportunity presented by crowdsourcing to gather a large quantity of command samples from a wide geographical area and to cover a range of languages and dialects without a high degree of resource requirement and scaling costs seems clear. Methods to overcome the challenges implicit in the use of such a method i.e. to communicate an action to study participants without conditioning choice of command phrase and to generate a high expectation of a command processing system have been demonstrated by studies performed in this thesis.

The use of crowdsourcing as a method of gathering command samples for automotive voice interfaces is therefore supported.

Further research to assess the effect of driving and crowdsourcing survey mindsets and the opportunity to improve the crowdsourcing method to minimise this provide the opportunity for further research.

### 9.1.10 Relating findings to literature

The findings of this thesis relate to literature reviewed in chapter 2 in a number of ways. In some cases, the findings support or extend existing material. In other cases findings suggest how opportunities potentially arising from the use of voice interfaces might be realised.

Graham and Carter [3, p.155] noted that "speech modality is both hands-free and eyes-free, meaning that the hands can be kept on the wheel and the eyes on the road". This is not the case however where say-what-you-see is employed in an automotive context to mitigate a low habitability, constrained command phrase system. In this case, a screen prompt shows the driver a set of viable voice commands. This requires eyes-off-road time to note the available commands and therefore does not realise (all of) the safety benefits of automotive voice interfaces. The methods proposed in this thesis do not utilise a display screen or require eyes off road time (except possibly the pressing of a voice interface activation control) and therefore address a benefit of voice interfaces as stated by Graham and Carter.

Operation of a voice interface requires selection of a command phrase. In cases where the system has high habitability, as the requirement to take an action is noted, the words used to note the action can simply be spoken as they occur. In the case of a low habitability system, one the action has been chosen, the user's natural choice of command phrase must be set aside and a command phrase which is acceptable to the system must be recalled from memory (or guessed based on experience of the language acceptable to the voice interface). This additional cognitive task may contribute to the cognitive load and potential effect on driving performance resulting from speech-based interaction with in-vehicle information systems noted by Lee et. al. [27, p.9]. The methods described in this thesis are primarily intended to engender natural language commands (and therefore higher habitability) but also may improve the potential for users (whether prompted or as a primary strategy) to easily guess acceptable command phrases.

A fully habitable voice interface could replace physical controls. In this case, speech would become the preferred medium as described by Bradford [31, p.63]. Replacing physical and touch-screen controls with a voice interface addresses the concern of Burnett and Porter [4, p.523] relating to difficulty in quickly locating and operating a control (and consequent eyes off road time) and the general point made by Graham and Carter [3, p.155] relating to limited

'dashboard real estate'. It would also address the need to “reduce response complexity, which in turn may reduce driver workload" noted by Gellatly [30, p.2]. Whilst the methods identified in this thesis do not result in a fully habitable system nor a fully syntactically habitable system (i.e. one which is able to interpret all command phrases which a user may employ), that the proportion of commands correctly processed can exceed the target set by users suggests a greater potential use of a voice interface in place of available physical controls. Once sufficiently established, it may be possible to move some physical controls to touch screen menus as a back-up modality in cases where a user is unable to use voice control.

The response threshold method described in study 6 elicits study participants choice of optimal confidence levels and therefore bases voice interface responses on user preference. However, driving performance is an additional measure noted by Kun et. al. [29, unformatted] who found that lower recognition accuracy resulted in higher steering input (to maintain a similar accuracy of lane position) and therefore potentially worse driving performance. Whether voice interface response thresholds which users prefer align with those which engender optimal driving performance may require further study.

A key focus of this thesis is the habitability of a system which bases the interpretation of spoken commands on a corpus of typed command examples (which can therefore be obtained by crowdsourcing). In the case of a pattern-matching voice interface, habitability in terms of proportion of commands recognised is not a fixed system value, it is instead affected by the specific user and the action required. Some users may align well with the choice of command phrases used by people from a similar geographic region, others may significantly differ. In the latter case, post-usage optimisation (shown in figure 9.2) may mitigate this difference, in other cases not. Section 6.3 shows that different actions result in different proportions of commands successfully interpreted. Hone and Baber [5, P.637] describe habitability as "match between the language people employ when using a computer system and the language that the system can accept". It may be useful to clarify that habitability from a user perspective can vary between systems and also between user and actions.

Baber [34, unformatted] asks whether it is natural to speak to computers and Bradford [31, p.63] suggests that human/human communication may not result in the correct syntax for a speech command interface, noting that "Human/computer communication generally consists of one syntactically complete utterance (roughly corresponding to a sentence) from the user followed by some kind of response from the computer. Human/human communication is much more varied.” Results in this thesis found that whilst text and spoken modality command phrases exhibit some of this difference (where typed is more verbose and perhaps more aligned to the human-human speech and spoken is more terse in the manner of human-machine speech), the effect on spoken command interpretation using typed command examples in place of spoken command examples is minimal (section 6.4).

That spoken and typed commands differed in choice of command phrase shows that at least one and potentially both differ from human-human language. This agrees with Bradford [31, p.65] who notes that 'natural language' in an HMI context may not necessarily be the same natural langue used in a human-human interaction and also Amalberti et. al. [9, p.4] who note a range of differences between human-human speech and human-machine speech.

In section 2.4.3, it was noted that Bigot et. al. [28, p.1051] found that "utterances were longer for speaking than for typing". However, the findings of studies 2 and 3 in this thesis ask participants to say or type commands which they would say to a voice interface (including voice recognition and command processing) and this resulted in longer utterances when using typed modality. This finding aligns with Baber et. al. [7, p.56] who noted that use of speech recognition resulted in reduced command length.

There were several instances where care was needed to avoid variability in coding similar word tokens, particularly when expressed differently between the two modalities. One example was the term halfway which was transcribed as 'half way' (two separate words) on occasion. Another was the number 22 which could be typed as twenty-two. This concurs with the findings of Zoltan-Ford [18, p.538].

Goodkind and Rosenberg [36, p.88] noted that typing was 'daunting for novice typists' and this could have resulted in a correlation between typing experience and commonality of typed and spoken modality command phrases. However, figure 4.8 (section 4.1.3) does not show correlation between mean typing speed and commonality of words typed and spoken. One explanation for this may be provided by Alves et. al. [84] who note that "Although motor execution is more demanding for slow typists, this higher demand neither prevented them from activating high-level processes concurrently with typing, nor changed the distribution of occurrences of the writing processes."

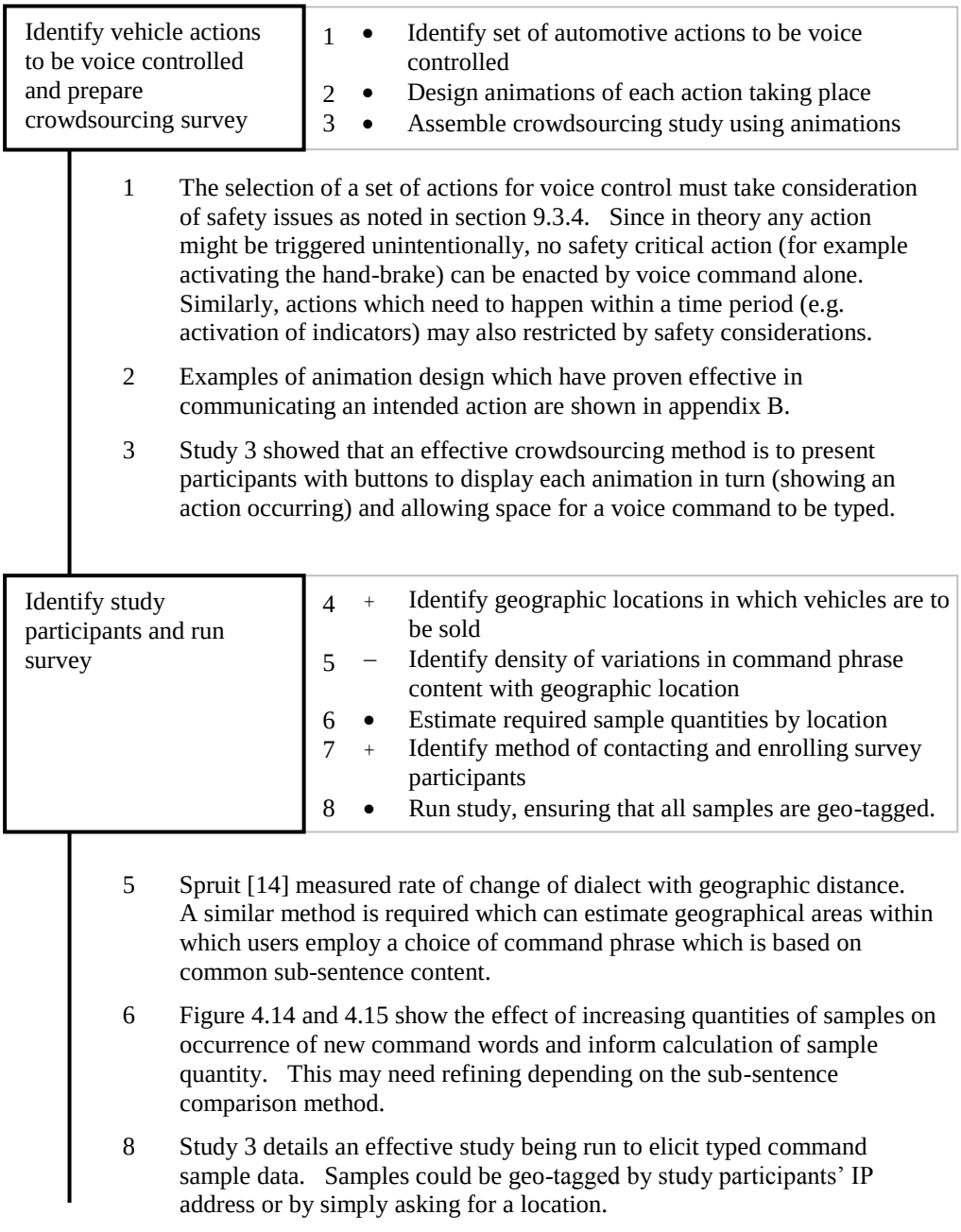
A question in study 1 asked "If you tried to use a new command, do you think the system would understand you?" The question could be extended to assess the likelihood of choosing a command the system can accept on a second attempt (when a user is prompted to rephrase their command). The methods proposed in this thesis may increase the likelihood that command rephrasing will succeed due to the similar origin of those who provided the corpus samples that that of the driver. If this were the case, it would make positive use of the effect noted by Baber et. al. [55, p.41] that (under stress), vocabulary violations could occur because "users revert to more familiar words and phrases" (which is potentially problematic for low habitability systems). The increased potential to successfully guess at or revert (whilst under stress) to a command phrase which the system can interpret could be argued to result in a more habitable system than continued failure on repeated attempts. Hone and Baber [5, P.637]

define habitability as "match between the language people employ when using a computer system and the language that the system can accept". Whilst this encompasses the language to which people may revert under stress or which they may select if their initial (natural) choice is not successful, it may be useful to note the additional circumstance.

### 9.1.11 Overview of proposed methodology

The following annotated flow diagram (figure 9.2) shows a proposed sequence of operation for an automotive voice interface based on the methods developed in this thesis. This includes corpus data collection and pre-processing, interface installation, pre-processing for a specific driver, live operation and accuracy optimisation.

Key:
+ Pre-existing methods
• Methods considered within this thesis
– Methods which require further research



Note: the following two steps will only be required if it is established that confidence thresholds for preferred voice interface response vary with geographical region. Otherwise levels, once established, can be applied in all cases.

| | | |
|---|---|---|
| Generate command processing action-models by geographic area | 9 • | Select a geo-tagged sample-set for each geographic area which has specific command-phrase content. |
| | 10 • | Run cross-validation process for each geographic area to prepare voice interface fingerprint models. |

9 Using the density of command phrase content variation identified in step 5, a set of command samples is selected for each area which has a discrete choice of command phrase content.

10 The cross-validation process described in section 5.5 is run on each set of samples, generating expected fingerprint patterns for each action.

| | | |
|---|---|---|
| Prepare and run response thresholds crowdsourcing survey | 11 • | Assemble a response threshold survey for each geographic area using previously designed animations and voice interface fingerprint models |
| | 12 • | Run surveys for each area. |

11 Chapter 8 describes a suitable crowdsourcing threshold survey method for eliciting response threshold levels. The animations developed in step 2 and the command samples crowdsourced in step 8 are used for this.

12 The process stated in step 11 is repeated and surveys are deployed for each area which has discrete command phrase content.

| | | |
|---|---|---|
| Prepare vehicle | 13 + | Devise voice activation method |
| | 14 + | Assemble vehicle with voice interface pre-programmed with geo-tagged crowdsourced sample-set and geo-tagged crowdsourced response thresholds. |

Note: The following step is required when the vehicle is first purchased.

| | | |
|---|---|---|
| Prepare voice interface for specific owner | 15 – | Identify the geographic area which has influenced new owner's choice of command phrase. |
| | 16 – | Establish driver identification |
| | 17 • | Select geo-tagged sample-set relevant to driver |
| | 18 • | Run cross-validation process to prepare voice interface fingerprint models. |

15 Identifying the location which has influenced the driver's choice of command phrase could be achieved by asking the driver for the information or initially by using the location from which the car was purchased. Further study is needed to ascertain the best method.

16 A method of identifying a driver and detecting a change in driver is needed. Potential options include using voice print, weight and/or seat position. Alternatively, the voice interface could state the pre-set driver when the car is started and allow a new driver to identify themself.

17 Since the samples gathered using the crowdsourcing survey in step 8 are geo-tagged, a set of samples can be selected which match the location which has influenced the driver's choice of command phrases.

18 Step 10 is repeated for the specific driver.

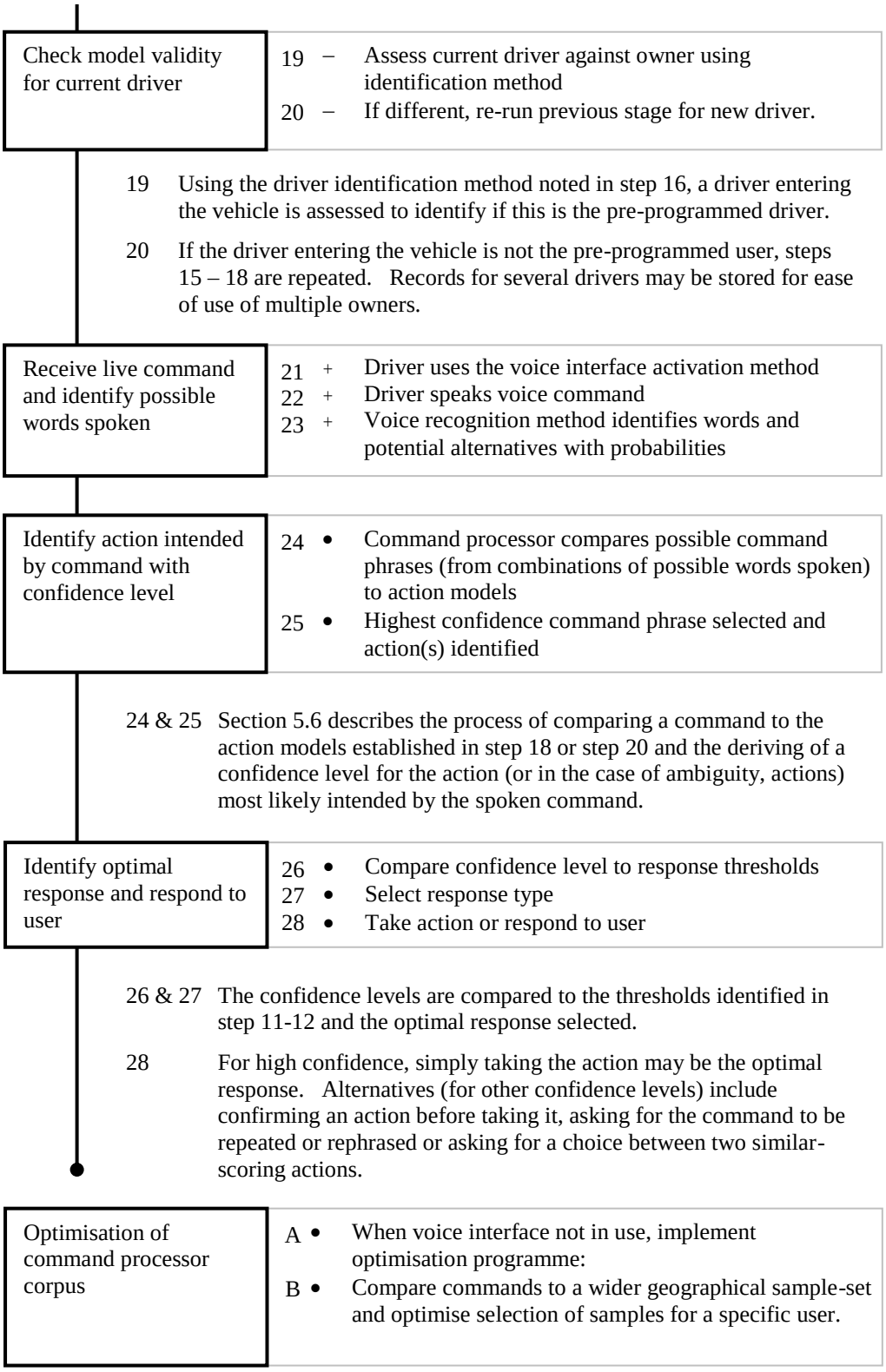



Figure 9.2. Annotated flow diagram for proposed methods

## 9.2 Conclusions

### 9.2.1 Sample gathering methods (research questions 1 & 2)

Research question 1: (Method of gathering command sample data) To investigate the potential to develop a crowd-sourcing method for gathering geographically tagged, automotive voice interface command samples such that effect on choice of vocabulary is minimised.

Research question 2: (Validity of command sample data) To assess the degree to which choice of vocabulary varies between a spoken modality (which would be used to operate an automotive voice interface) and a typed modality (which facilitates crowdsourcing).

Studies 2 & 3 assessed methods by which command samples might be gathered using crowdsourcing to overcome the scaling costs arising when considering many samples over a wide geographical area. The results of study 2 showed limited typed and spoken modality effects on command phrasing and also highlighted the need for care in selecting a prompting method to avoid ambiguity. Study 3 used an improved prompting method and was thereby able to obtain samples with all participants addressing a common subject and action. The use of typed samples in place of spoken samples incurred no measured reduction in action-identification (compared to the use of spoken samples) when processing spoken commands. Results therefore support the collection of geographically distributed command samples by using a typed modality crowdsourcing method, thereby avoiding scaling of the required resources.

### 9.2.2 Command processing method development (research question 3)

Research question 3: (Method of command processing using command sample data) To assess how effectively spoken commands may be interpreted using only crowdsourced example commands, focussing on a simple token-based command processing approach.

As the data available to identify the action intended by natural-language commands was constrained to a set of command examples obtained via crowdsourcing, the method developed was a simple pattern matching empirical model. Whilst relatively simple in approach, within the constraints of the set of actions selected and for commands given by the set of participants, the command processing method was able to correctly identify the intended action for 87.1% of spoken commands when using a typed-modality corpus. Whilst this is slightly lower than the level of action identification desired by participants of the semi structured study performed in study 1, the gap is minimal and increasing sample-count for actions which engender more verbose commands would bring the performance to the desired level. Further options to improve the method described in chapter 5 are noted in section 9.3.6.

### 9.2.3 Robustness to voice recognition errors (research question 4)

Research question 4: (Habitability of command processing method) To evaluate the degree of success and confidence with which the command interpretation method is able to identify actions intended by fully and partially recognised commands when both the command and the crowdsourced command samples originate from similar geographic locations.

Using the command processing method developed in chapter 5, it was found that action identification increased as number of corpus samples increased and the degree of improvement reduced with successive samples in an approximately logarithmic pattern. Robustness of the command processing method and crowdsourced, typed commands was evaluated by assessing spoken commands with a single misrecognised word. Whilst lower-level effects were noted, no change was found in the proportion of commands correctly identified. Results show that basing corpus sample quantity on command word-count and that pattern-matching all command words provides robustness in cases where a single word is not recognised by the voice recognition method.

### 9.2.4 System response confidence thresholds (research question 5)

Research question 5: (Method of gathering response threshold data and effect on command processing) To investigate the potential to develop a crowd-sourcing method of obtaining geographically tagged threshold levels for voice interface system responses to optimally manage differing confidence levels when identifying an intended action.

Study 6 provided participants with the opportunity to set response thresholds to tune system responses according to their preference. The study found initially that the ‘confirm before acting’ to ‘act immediately’ threshold was not common for all participants. However, further investigation showed that a subset of participants would prefer to have their commands either immediately acted on or to simply re-state the command (i.e. not to have the system ask for confirmation). Separating this group enabled a single response threshold level to be identified for each type of threshold which would satisfy the set of study participants. Results show that crowdsourcing is a practical way to obtain user-optimised threshold settings and that it was possible to select threshold settings which are acceptable to all users. Results also show that a subset of users prefers to have an option to disallow confirmation and disambiguation dialogues.

## 9.3 Further research opportunities

This thesis has considered many of the methods which would be needed to implement a voice interface based on crowdsourced data and a pattern-matching token-based command interpreter and to optimise the human-machine interface to best suit system users. The studies performed were scoped to a set of actions selected to evaluate aspects of command processing performance and to a set of participants who worked for the same large company and might therefore be expected to employ a similar choice of command phrasing. One option for further research is in extending the studies performed in this thesis to assess a wider range of actions and to consider groups of people with different languages and dialects. Other opportunities for further research divide broadly into investigation of the applicability of methods in untested circumstances (for example, during live or simulated driving and whilst under time stress) and also in developing additional methods which would be required before a voice interface could be realised (for example, driver identification). Possible areas for additional research are described in sections 9.3.1 to 9.3.6.

### 9.3.1 Human-machine interface and human factors

For voice interfaces which approach human capability, a system which is constantly listening to the user's speech and identifying when a command is being spoken may be preferable. Where this may result in non-command words being inadvertently interpreted as such, a positive user activation of the voice interface may be needed to prevent this. Options include physical button(s) or speaking a specific word or phrase. Oh et. al. [85] presents the potential to activate a voice interface by detecting a user glancing at it, although an automotive system may lack a definitive feature to glance at. Opportunities for future research includes the basis for deciding the benefits of each voice interface activation modality and type, the design and location of a physical button and/or the choice of system-activation word or phrase. Another option for research is into social acceptance of drivers speaking a command phrase to activate a voice interface.

When first presented with a voice interface which is intended to understand natural language commands, a user may present the system with unexpected commands such as 'hello' or 'how are you' and user-perception of voice interface command interpretation capability may depend on responses. User interactions with 'chatbots' such as the ALICE system described by Shah [86] and a range of other systems described by Shawar and Atwell [87] show human willingness to interact with a computer in a frivolous fashion. If a poor response to a frivolous initial question were to result in poor perception of voice interface confidence, commands may thereafter be phrased in an attempt to mitigate perceived limited capability. Commands issued with expectations of poor system capability may be poorly identified when using a corpus comprising samples from a context expecting good system capability. In such cases, the expectation of poor performance could become self sustaining. One option to mitigate poor recognition of frivolous initial commands might be to study such interactions and pre-set

responses, thereby maintaining or creating high user expectations and hence eliciting commands which are in the same high-expectation context as those in which a command corpus may be gathered (resulting in improved pattern matching). Another aspect of first use interaction with a voice interface relates to possible trepidation or other emotional response to 'talking to an intelligent computer' as described by Rosen et. al. [88]. Methods for introducing users to natural language voice interfaces are a broad area for future research.

In study 6 (chapter 8), on-screen text responses were worded to simply communicate the results of commands given. An option for further research involves the effect of voice interface response phrases, gender, frequency, tone and emphasis and rate of speech. Fixed phrase responses or responses calculated to match user syntax are possibilities, however it would be necessary to preclude nonsensical responses which can result from generation by algorithm as noted by Shawar and Atwell [89, p.38]. Study 6 also found that a subset of users would prefer not to receive confirmation / disambiguation requests and there is potential for further research on factors affecting this preference and into interface design which allows users to choose levels of system dialogue.

### 9.3.2 Crowd-sourcing and information source development

Gathering command samples using a crowdsourced, typed modality method requires finding and engaging people who are willing to provide the command samples. Participants may be persuaded to assist by offering material reward or the potential to win material reward (for example, in a prize draw). Smith et. al. [90, p.24] describe the provision of material reward as an extrinsic motivation (where an inducement is provided which is not related to the study) and describe a study method which provides enjoyment to the participant as an intrinsic inducement. One option for an intrinsic inducement may be online games which are able to offer non-textual action prompts (or perhaps indirect context-related 'prompts') and elicit natural language commands. Data could be gathered from many countries and cultural factors may need consideration. Methods for gathering crowdsourced command samples are an area for further research.

One option for improving natural language voice interface habitability with respect to the real-world usage context is to gather command samples during live usage. It may be necessary to categorise live sample gathering as equivalent to an interview where, as described by Orb et. al. [91, p.94], 'an interview is usually equated with confidentiality, informed consent and privacy'. The use of samples gathered during live usage could be constrained to be used only by the person who gave the commands, thus avoiding privacy issues. Gathering samples for use by other users may however invoke privacy issues and one option could be to employ an encryption method such that the corpus could be used for command identification but could not be extracted. Initial consent to gather data may need to be supplemented with the option to later decline if, for example, a user unintentionally provides the interface with speech which is

confidential and not related to the command interface. It may also be necessary to allow a user to withdraw previously provided command samples and in this case the consequent requirement to tag samples with a user identifier would introduce both an administrative burden and raise further privacy concern. User-acceptability of in-usage automotive command sample recording may need research.

One requirement of the simple command processing method developed in chapter 5 is that the command samples are given by people whose choice of command phrasing is similar to that of a potential user. To ensure this is the case, methods are needed which build upon the findings of Spruit [14] such that the number of command samples gathered can be decided according to how rapidly command phrasing varies with geographical distance. For each user, it may be possible to optimise the selection of command samples of which the corpus comprises at times when the system is not in-use by re-processing user commands with variants of the command sample corpus and searching for higher confidence. A search algorithm to achieve this may be a basis for research.

The animation-based prompting method used in study 3 (section 4.2) avoided including words in the animations as this could have biased participants' choice of command phrase. This may be difficult in the case of some actions where the animation alone does not fully communicate the intended action (for example, loading the in-car entertainment with a play-list of music would be difficult to convey without words). The effect on command phrases of using words in such cases and options to avoid doing so (if necessary) are an opportunity for further research.

### 9.3.3 Isolation and usage of data within a command

Voice commands may include data. For example, data may describe the direction or degree of a desired action. In these cases, the data statement may need to be extracted from the command before it can be quantified. Identifying the value within data may also need consideration; for example a command to open a window 'a little', 'a crack' or 'ten percent' may intend the same quantity. A potential way to respond to qualitative data statements may be to observe user responses to adjustments and use this information to optimise quantitative interpretations. For example, if a command stated 'increase music volume' and after an adjustment, the command was repeated, future such commands might engender a greater increase. It may be possible to use the method developed in chapter 5 to extract data-statements using a best-fit confidence approach and this and other approaches require further work.

### 9.3.4 Safety

Voice interface methods which provide improved habitability have the potential to encourage use of voice interfaces and thereby realise the safety benefits identified in material such as that reviewed in section 2.3.1. The use of natural language voice interface methods may also affect safety in cases where commands are not understood and particularly where an unintended action is substituted. The resulting delay could preclude voice interface usage for actions where speed of interface response is safety-related. Actions of this type might include operating the vehicle indicators or applying the handbrake whilst stationary (to prevent vehicle roll-back). The potential for an unintended action to be inadvertently substituted may disallow provision of a speech interface for any action which could, if enacted at an inopportune moment, cause a safety issue. For example, the inadvertent application of a vehicle handbrake while driving would not be desirable. Whilst safety issues could prevent a number of actions being available via voice interfaces, the use of automatic context and/or additional user-confirmation checks could reduce the circumstances where this applies (e.g. a maximum allowable vehicle speed for handbrake application). Also, some actions may be possible in one direction (e.g. turning on the headlights) but not the other direction (unless a system detected sufficiently high ambient light to 'allow' the command). Directional and/or context-based command constraints may cause confusion to users. Safety-related constraints on voice interface functionality and mitigation of consequent effects on user satisfaction require consideration.

### 9.3.5 Effects of natural language command interfaces on user satisfaction

Study 1 and study 6 both considered action-identification thresholds for optimal system responses, the former by asking survey participants their opinion and the latter by asking users to tune a command processor to their preference. When voice interface systems based on use of natural language and on confidence-based thresholds are in commercial use, one option for research is the development of an algorithm which evaluates user-reaction to system responses and further optimises thresholds settings. It may also be possible to assess user-acceptance of voice interfaces over time such that when a sufficiently high acceptance-level has been achieved and the voice interface has become the primary modality, conventional interface control-clutter can be reduced (for example by moving conventional buttons to a nested touch-screen menu).

### 9.3.6 Optimisation of command processing method

The command processing method developed in chapter 5 was less able to identify the intended action for two of the twelve actions evaluated, this occurred due to similarity of command phrases. Methods to optimise handling of ambiguous actions are an opportunity for research. Further research into how to improve habitability of the method could consider reintroducing more complex sub-sentence fragment (possibly using the binary method described in section

5.3.2) and the possibility of subdividing expected pattern matching models by grammatical word type. Vehicle context may be an opportunity for further research; this would include the circumstance in which commands are spoken when determining the probability of which action is intended. For example it is perhaps more likely that the driver may ask for the heating to be increased in the context of low cabin temperature.

As a driver uses a voice interface, the information provided could be used to improve habitability for the specific driver. For example, where commands are spoken and an action is taken, the command phrase could be added to the corpus of correct commands (for that action). This evolution of the corpus could result in a set of commands which are initially a limited match to a driver's command phrasing becoming more aligned and effective over successive uses. Evolving the corpus in this way may eventually allow the voice interface to interpret commands which would initially not be understood.

The command processing method developed in chapter 5 was tested on a specific set of study participants and involved a specific set of prompted actions. One option for further research is the testing of this method for sets of study participants from different regions and study participants who speak different languages and also the consideration of a wider range of actions. As demonstrated by study 3 (section 4.2), the samples needed for this research could be obtained by using a typed modality and by using a crowdsourcing method.

# 10 Bibliography


1 Estellés-Arolas, E. and González-Ladrón-de-Guevara, F. Towards an integrated crowdsourcing definition. Journal of Information science, 2012, 38(2), pp.189-200.

2 McCarthy, J. What is artificial intelligence. Technical report, 2007, Computer Science Department, Stanford University

3 Graham, R. & Carter. C. Comparison of speech input and manual control of in-car devices while on the move. Personal Technologies. 2000, 4(2-3), pp.155-64.

4 Burnett, G.E. & Porter. J.M. Ubiquitous computing within cars: designing controls for non-visual use. International Journal of Human-Computer Studies 2001, 55(4), pp.521–531.

5 Hone, K.S. and Baber, C. Designing habitable dialogues for speech-based interaction with computers. International Journal of Human-Computer Studies, 2001, 54(4), pp.637-662.

6 Bobrow, D. G. Natural Language Input for a Computer Problem Solving System. Ph.D. Thesis, Mathematics Department, M.I.T., Cambridge, Mass. 1964.

7 Baber, C., Johnson, G. I., & Cleaver, D. Factors affecting users' choice of words in speech-based interaction with public technology. International Journal of Speech Technology, 1997, 2(1), pp.45–59.

8 Baber, C. and Noyes, J. Automatic speech recognition in adverse environments. Human factors 1996, 38(1), pp.142-155.

9 Amalberti, R., Carbonell, N. and Falzon, P. Users' representation of the machine. Rapports de Recherche, 1989, 1125(8).

10 Furnas, G.W., Landauer, T.K., Gomez, L.M. and Dumais, S.T. The vocabulary problem in human-system communication. Communications of the ACM, 1987, 30(11), pp.964-971.

11 Hone, K.S. and Baber, C. Modelling the effects of constraint upon speech-based human–computer interaction. International Journal of Human-Computer Studies, 1999, 50(1), pp.85-107.

12 Watt, W.C. Habitability. Journal of the Association for Information Science and Technology, 1968, 19(3), pp.338-351.

13 Juang B.H., Rabiner L.R. Automatic speech recognition–A brief history of the technology development. Encyclopaedia of Language and Linguistics. 2005.

14 Spruit, M.R. Measuring Syntactic Variation in Dutch Dialects. Literary and linguistic computing, 2006, 21(4), pp.493-506.

15 Ward, N.G., Rivera, A.G., Ward, K. & Novick, D.G. Root causes of lost time and user stress in a simple dialog system. Proc. of Interspeech, 2005.

16 McTear, M.F. Spoken dialogue technology: enabling the conversational user interface. ACM Computing Surveys (CSUR), 2002, 34(1), pp.90-169.

17 Zudilova, E.V., Sloot, P.M. and Belleman, R.G. A multi-modal interface for an interactive simulated vascular reconstruction system. In Multimodal Interfaces, 2002. Proceedings. Fourth IEEE International Conference on, pp. 313-318.

18 Zoltan-Ford, E. How to get people to say and type what computers can understand. International Journal on Man-Machine Studies, 1991, 34, pp.527-547.

19 Dudley, H., Riesz, R.R. & Watkins, S.S.A. A synthetic speaker. Journal of the Franklin Institute, 1939, 227(6), pp.739-764.

20 Alm, H. and Nilsson, L. Changes in driver behaviour as a function of hands free mobile phones—a simulator study. Accident Analysis & Prevention, 1994, 26(4), pp.441-451.

21 Baum, L.E., Petrie, T., Soules, G. and Weiss, N. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. The annals of mathematical statistics, 1970, pp.164-171.

22 Furnas, G.W., Landauer, T.K., Gomez, L.M. and Dumais, S.T. Human factors and behavioural science: Statistical semantics: Analysis of the potential performance of key-word information systems. Bell System Technical Journal, The, 1983, 62(6), pp.1753-1806.

23 Greenberg, J., Tijerina, L., Curry, R., Artz, B., Cathey, L., Kochhar, D., Kozak, K., Blommer, M. & Grant, P. Evaluation of Driver Distraction Using an Event Detection Paradigm. Journal of the Transportation Research Board, 2003, 1843, pp.1-9.

24 Parkes, A.M. & Hooijmeijer, V. Driver situation awareness and carphone use. Proceedings of the 1st Human-Centered Transportation Simulation Conference, 2001

25 Strayer, D.L., Drews, F.A., Albert, R.W. and Johnston, W.A. Cell phone induced perceptual impairments during simulated driving, 2001, Driving Assessment, pp.14-19.

26 Burnett, G.E., Summerskill, S.J. & Porter, J.M. On-the-move destination entry for vehicle navigation systems: Unsafe by any means?. Behaviour & Information Technology, 2004, 23(4), pp.265-272

27 Lee, J.D., Caven, B., Haake, S. & Brown, T.L. Speech-based interaction with in-vehicle computers: The effect of speech-based email on driver's attention to the roadway, Human. Factors, 2001, 43, pp.631–640.

28 Le Bigot, L., Rouet, J.F. & Jamet, E., 2007. Effects of Speech- and Text-Based Interaction Modes in Natural Language Human-Computer Dialogue. Human Factors: The Journal of the Human Factors and Ergonomics Society, 2007, 49(6), pp.1045–1053.

29 Kun, A.L., Paek, T. and Medenica, Z. The Effect of Speech Interface Accuracy on Driving Performance. Interspeech, 2007, pp.1326-1329.

30 Gellatly, A. W. The use of speech recognition technology in automotive applications. Doctoral dissertation, Virginia Polytechnic and State University. 1997.

31 Bradford, J.H. The human factors of speech-based interfaces: A research agenda. ACM SIGCHI Bulletin. 1995, 27(2), pp.61-67.

32 Harris, G., Begg, I. & Upfold, D. On the role of the speaker's expectations in interpersonal communication. Journal of Verbal Learning and Verbal Behavior, 1980, 19(5), pp.597-607.

33 Edlund, J., Gustafson, J., Heldner, M. and Hjalmarsson, A. Towards human-like spoken dialogue systems. Speech Communication, 2008, 50(8-9), pp.630-645.

34 Baber, C. Human factors aspects of automatic speech recognition in control room environments. Systems and Applications of Man-Machine Interaction Using Speech I/O, IEE Colloquium on, 1991, pp.10/1-10/3.

35 Luximon, A., Duffy, V.G. and Zhang, W. Performance differences in a cross-cultural comparison of voice enhanced interface. International Journal of Industrial Ergonomics, 2001, 28(3), pp.133-142.

36 Goodkind, A. and Rosenberg. Muddying the multiword expression waters: How cognitive demand affects multiword expression production. In Proceedings of NAACL-HLT, 2015, pp. 87-95.

37 Dahlbäck, N. and Jönsson, A. Empirical studies of discourse representations for natural language interfaces. Proceedings of the fourth conference on European chapter of the Association for Computational Linguistics, 1989, pp.291-298.

38 Ai, H., Raux, A., Bohus, D., Eskenazi, M. and Litman, D.J. Comparing spoken dialog corpora collected with recruited subjects versus real users. Proc. of SIGdial, 2007.

39 Gould, J.D. How to design usable systems. Readings in Human Computer Interaction: Towards the Year, 1988, p.93-121.

40 Ogden, W.C. and Bernick, P. Using natural language interfaces. Handbook of human-computer interaction, 1997, pp.137-161.

41 Schmandt, C. Voice Communications with Computers. 1994, Thomson Learning.

42 Green, A., Eklundh, K.S., Wrede, B. and Li, S. Integrating miscommunication analysis in natural language interface design for a service robot. In Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on, pp. 4678-4683.

43 Magerman, D. Natural Language Parsing as Statistical Pattern Recognition. Ph.D. thesis, Stanford University, 1994.

44 Hulth, A. Improved automatic keyword extraction given more linguistic knowledge. Proceedings of the 2003 conference on Empirical methods in natural language processing, Association for Computational Linguistics, 2003, pp. 216-223.

45 Cambria, Erik, and Bruce White. "Jumping NLP curves: a review of natural language processing research (review article)." Computational Intelligence Magazine, IEEE 9.2, 2014, pp.48-57.

46 Erkan, G. and Radev, D.R. LexRank: Graph-based lexical centrality as salience in text summarization. Journal of Artificial Intelligence Research, 2004, pp.457-479.

47 Mihalcea, R. & Tarau, P. TextRank: Bringing order into texts. Association for Computational Linguistics, 2004.

48 Kwasny, S.C. and Sondheimer, N.K. Relaxation techniques for parsing grammatically III-formed input in Natural Language Understanding Systems. Computational Linguistics, 1981, 7(2), pp.99-108.

49 Liu, H., Lieberman, H. and Selker, T. January. A model of textual affect sensing using real-world knowledge. Proceedings of the 8th international conference on Intelligent user interfaces, ACM, 2003, pp.125-132.

50 Behrend, T.S., Sharek, D.J., Meade, A.W. & Wiebe, E.N. The viability of crowdsourcing for survey research. Behavior research methods, 2011, 43(3), pp.800-813.

51 Hua, Z. and Ng, W.L. Speech recognition interface design for in-vehicle system. Proceedings of the 2nd International Conference on Automotive User Interfaces and Interactive Vehicular Applications. ACM, 2010, pp.29-33.

52 Bohus, D. and Rudnicky, A.I. Sorry, I Didn't Catch That!. Recent Trends in Discourse and Dialogue, Springer Netherlands, 2008, pp.123-154.

53 Kamm, C. User interfaces for voice applications. Proceedings of the National Academy of Sciences, 1995, 92(22), pp10031-10037.

54 Boyce, S. and Gorin, A. User interface issues for natural spoken dialog systems. Proc. ISSD, 1996, 96, pp.65-68.

55 Baber, C., Mellor, B., Graham, R., Noyes, J. M. & Tunley, C. Workload and the use of automatic speech recognition: The effects of time and resource demands. Speech Communication, 1996, 20(1), pp.37–53.

56 Chen, F. and Sääv, J. The effect of time stress on automatic speech recognition accuracy when using second language. INTERSPEECH, 2001, pp. 2737-2740.

57 Bohus, D. Error awareness and recovery in conversational spoken language interfaces. Doctoral dissertation. SRI International. 2007

58 Boyce, S. and Gorin, A. User interface issues for natural spoken dialog systems. Proc. ISSD, 1996, 96, pp.65-68.

59 Graham, Robert, and Chris Carter. Comparison of speech input and manual control of in-car devices while on the move. Personal Technologies, 2000, 4(2-3), pp.155-164.

60 McTear, M.F. Spoken dialogue technology: enabling the conversational user interface. ACM Computing Surveys (CSUR), 2002, 34(1), pp.90-169.

61 Lo, V.E.W. & Green, P.A. Development and evaluation of automotive speech interfaces: useful information from the human factors and the related literature. International Journal of Vehicular Technology, 2013.

62 Oulasvirta, A., Engelbrecht, K.P., Jameson, A. & Möller, S. The relationship between user errors and perceived usability of a spoken dialogue system, 2006, ISCA/DEGA.

63 K. S. Hone and R. Graham. Towards a tool for the subjective assessment of speech system interfaces (SASSI). Natural Language Engineering, 2000, 6(3/4), pp.287-303.

64 Schuman, H. and Presser, S. Question wording as an independent variable in survey analysis. Sociological Methods & Research, 1977, 6(2), pp.151-170.

65 Zaller, J. and Feldman, S. A simple theory of the survey response: Answering questions versus revealing preferences. American journal of political science, 1992, pp.579-616.

66 Bertrand, M. and Mullainathan, S. Do people mean what they say? Implications for subjective survey data. The American Economic Review, 2001, 91(2), pp.67-72.

67 Elo, S. and Kyngäs, H. The qualitative content analysis process. Journal of advanced nursing, 2008, 62(1), pp.107-115.

68 Casali, S.P., Williges, B.H. and Dryden, R.D. Effects of recognition accuracy and vocabulary size of a speech recognition system on task performance and user acceptance. Human Factors: The Journal of the Human Factors and Ergonomics Society, 1990, 32(2), pp.183-196.

69 Schmidt, A., Beigl, M. and Gellersen, H.W. There is more to context than location. Computers & Graphics, 1999, 23(6), pp.893-901.

70 Chan, F.Y. and Khalid, H.M. Is talking to an automated teller machine natural and fun?. Ergonomics, 2003, 46(13-14), pp.1386-1407.

71 Antoniazzi, A. Storytelling and Videogames: an interdisciplinary approach to interactive multimedia models for children. Studi sulla formazione, 2013, 16(1), pp.67-76.

72 Pinto, J. and Sitaram, R.N.V. Confidence measures in speech recognition based on probability distribution of likelihoods. INTERSPEECH, 2005, pp.3001-3004.

73 Eysenck, M.W. and Keane, M.T. Cognitive psychology: A student's handbook. Taylor & Francis , 2000.

74 Wolters, M., Georgila, K., Moore, J.D. and MacPherson, S.E. Being old doesn't mean acting old: How older users interact with spoken dialog systems. ACM Transactions on Accessible Computing (TACCESS), 2009, 2(1), p.2.

75 Banko, M. & Brill, E. Mitigating the paucity-of-data problem: Exploring the effect of training corpus size on classifier performance for natural language processing. Proceedings of the first international conference on Human language technology research, Association for Computational Linguistics, 2001, pp.1-5.

76 Gale, W.A., Church, K.W. & Yarowsky, D. A method for disambiguating word senses in a large corpus. Computers and the Humanities, 1992, 26(5-6), pp.415-439.

77 Banko, M. & Brill, E. Scaling to very very large corpora for natural language disambiguation. Proceedings of the 39th annual meeting on association for computational linguistics, Association for Computational Linguistics, 2001, pp.26-33.

78 Jamson, S.L. and Jamson, A.H. The validity of a low-cost simulator for the assessment of the effects of in-vehicle information systems. Safety science, 2010, 48(10), pp.1477-1483.

79 Groeger, J.A., Carsten, O.M.J., Blana, E. and Jamson, A.H. Speed and distance estimation under simulation conditions. Vision in vehicles, 1999, 7, pp.291-299.

80 Letts, L., Wilkins, S., Law, M., Stewart, D., Bosch, J. and Westmorland, M. Guidelines for critical review form: Qualitative studies (Version 2.0). Ontario, Canada: McMaster University Occupational Therapy Evidence-Based Practice Research Group, 2007.

81 Speer, S.A. and Hutchby, I. From ethics to analytics: Aspects of participants' orientations to the presence and relevance of recording devices. Sociology, 2003, 37(2), pp.315-337.

82 Juang, B.H. and Rabiner, L.R. Automatic speech recognition–A brief history of the technology development. Encyclopedia of Language and Linguistics, 2005.

83 Shneiderman, B. Designing the user interface. Pearson Education India, 2002.

84 Alves, R.A., Castro, S.L. and Olive, T. Execution and pauses in writing narratives: Processing time, cognitive effort and typing skill. International journal of psychology, 2008, 43(6), pp.969-979.

85 Oh, A., Fox, H., Van Kleek, M., Adler, A., Gajos, K., Morency, L.P. and Darrell, T. Evaluating look-to-talk: a gaze-aware interface in a collaborative environment. CHI'02 Extended Abstracts on Human Factors in Computing Systems, ACM, 2002, pp.650-651.

86 Shah, H. ALICE: an ACE in Digitaland. tripleC: Communication, Capitalism & Critique. Open Access Journal for a Global Sustainable Information Society, 2006, 4(2), pp.284-292.

87 Shawar, B.A. and Atwell, E. Chatbots: are they really useful? LDV Forum, 2007, 22(1), pp. 29-49.

88 Rosen, L.D., Sears, D.C. and Weil, M.M. Computerphobia. Behavior Research Methods, Instruments, & Computers, 1987, 19(2), pp.167-179.

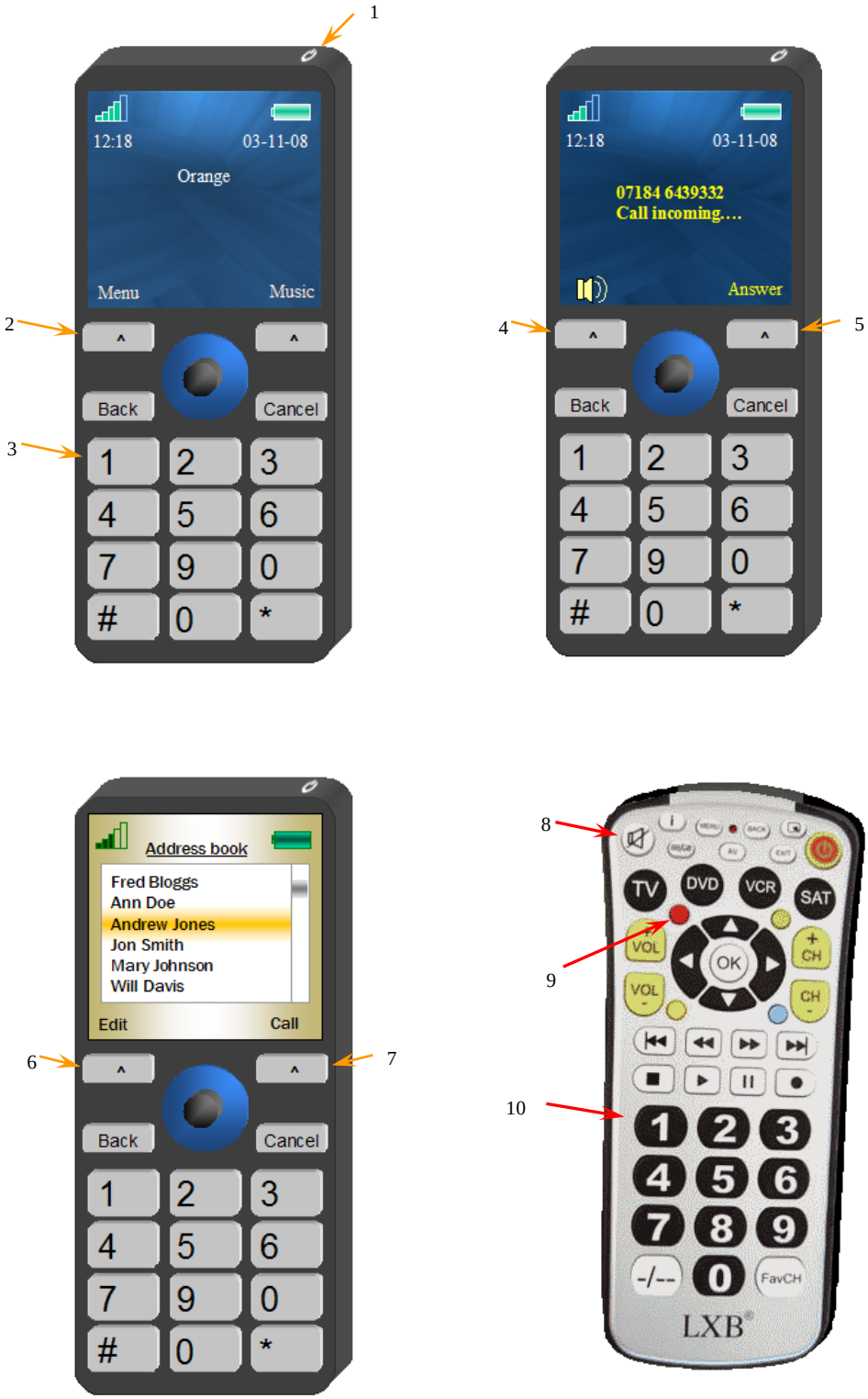
89 Shawar, B.A. & Atwell, E. Chatbots: are they really useful?. LDV Forum, 2007, 22(1), pp.29-49.

90 Smith, D., Manesh, M.M.G. and Alshaikh, A. How can entrepreneurs motivate crowdsourcing participants?. Technology Innovation Management Review, 2013, 3(2).

91 Orb, A., Eisenhauer, L. and Wynaden, D. Ethics in qualitative research. Journal of nursing scholarship, 2001, 33(1), pp.93-96.

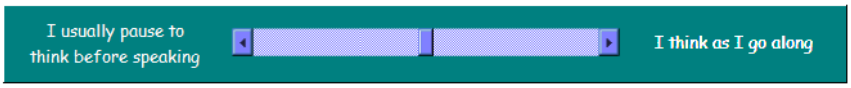
# *Appendix A*

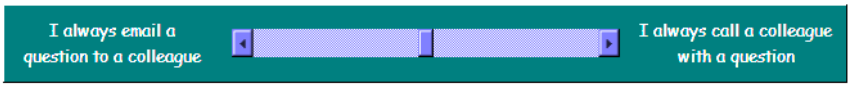
## Graphical prompts used for Study 2




Sprite
Coke

11


I usually pause to think before speaking
I think as I go along

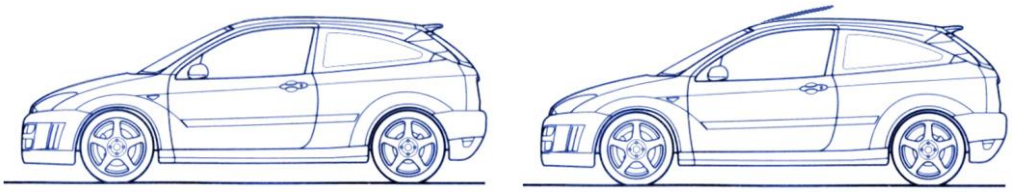

I always email a question to a colleague
I always call a colleague with a question

# *Appendix B*

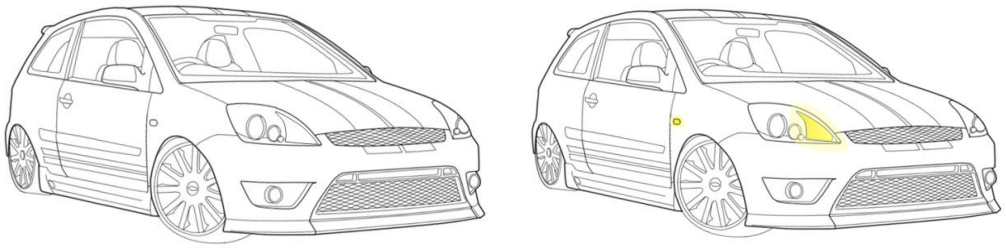
## Animated prompts used for Study 3

The following images show the start and end of the animation states or for cyclic operations such as indicators or windscreen wiper operation, show the two limit states.

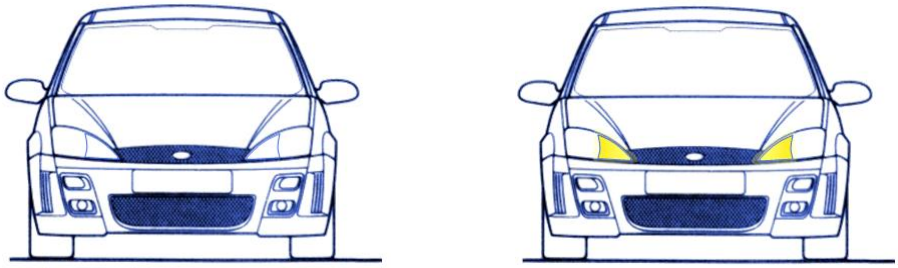
1.



2.

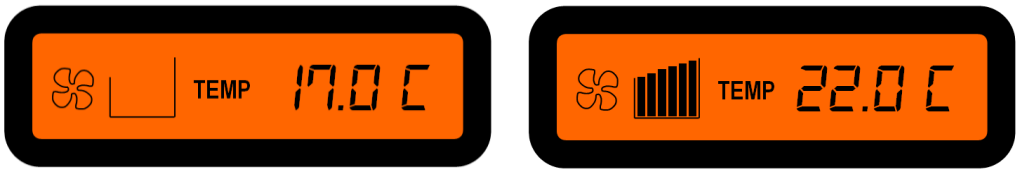


3.

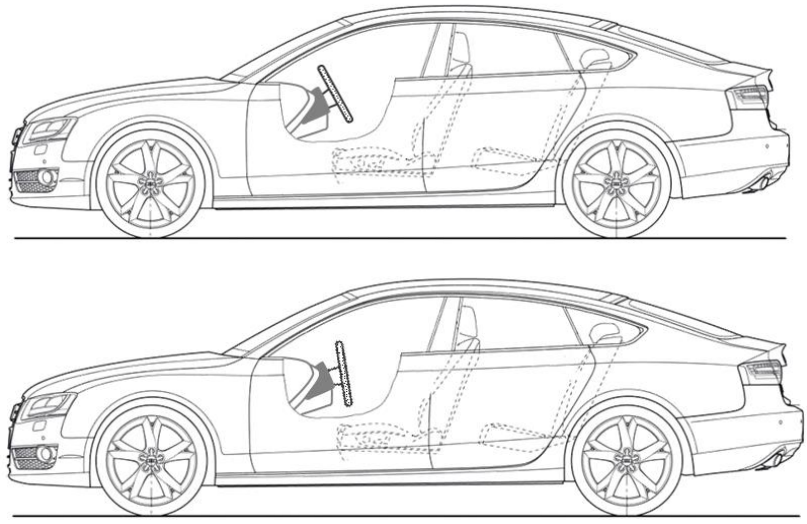


4.

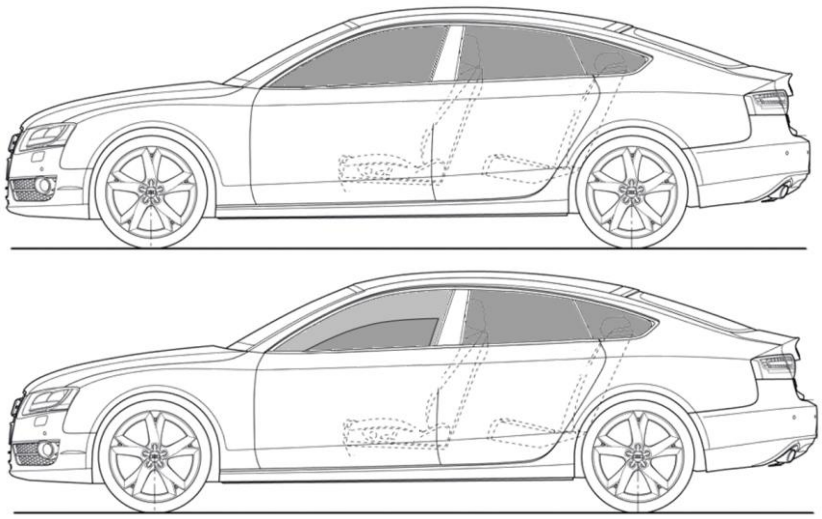

5.



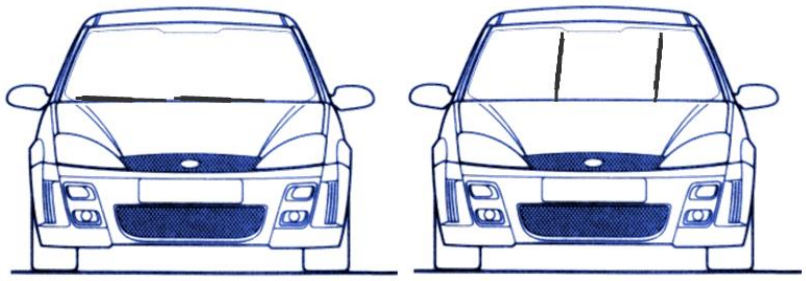
6.



7.



8.

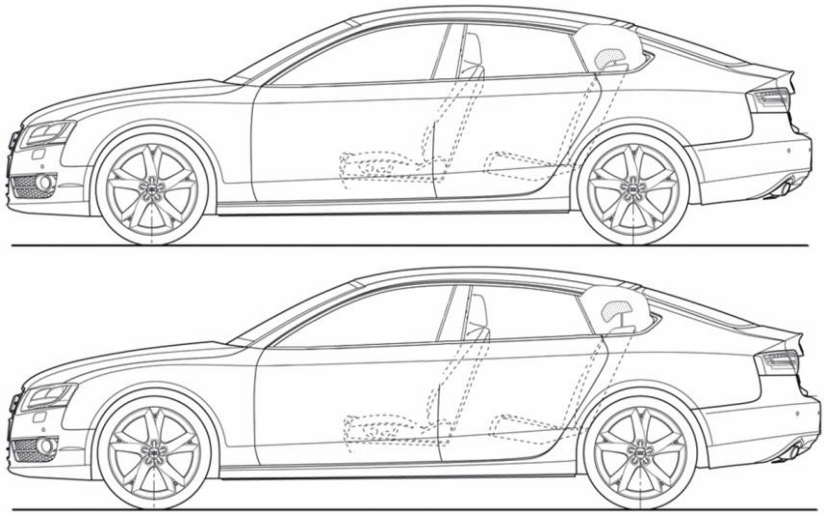


9.

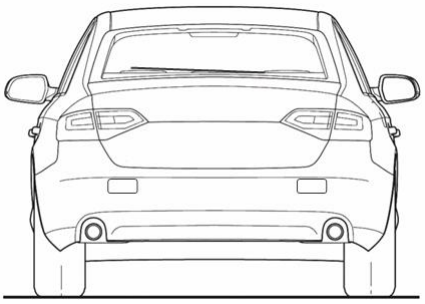

70
50
90
100 120
80
140
30
60
160
110
1/1
40
180
1/2
200
10
MPH
130
TRIP
ODO

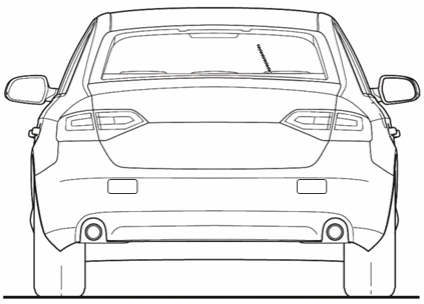

70
50
90
100 120
80
140
30
60
160
110
1/1
40
180
1/2
200
10
MPH
130
TRIP
ODO

10.



11.





12.