Intelligent Voice Navigation of Spreadsheets: An Empirical Evaluation

Derek Flood, Kevin Mc Daid, Fergal Mc Caffery, and Brian Bishop

Dundalk Institute of Technology, Dundalk, Co. Louth, Ireland derek.flood@dkit.ie, kevin.mcdaid@dkit.ie, fergal.mccaffery@dkit.ie and brian.bishop@dkit.ie

Abstract. Interaction with software systems has become second nature to most computer users, however when voice recognition is introduced, this simple procedure becomes quite complex. To reduce this complexity for spreadsheet users, the authors have developed an intelligent voice navigation system called iVoice. This paper outlines the iVoice system and details an experiment that was conducted to determine the efficiency of iVoice when compared to a leading voice recognition technology.

1 Introduction

Technological advances and a desire for mobility have substantially reduced the size of complex systems, calling into question the practicality of traditional input mechanisms. New mechanisms, such as voice recognition, are emerging that will allow for easier control of these new mobile systems.

Voice recognition technology will not only benefit users of mobile systems. Using voice recognition technology sufferers of RSI (Repetitive Strain Injury) can more comfortably control computer applications, which would otherwise require extensive use of the keyboard and mouse[1]. One such application is Microsoft Excel, the most common spreadsheet package. Spreadsheets are used for a wide variety of tasks from record keeping to financial statements and although spreadsheets are a ubiquitous software tool, in recent years their reliability has been shown to be very poor. Section 2 overviews the spreadsheet technology and ways to address this absence of reliability.

Voice control of software applications, such as spreadsheets, has been proven to be difficult[2]. Some software domains, as outlined in Section 3, have been adapted to take advantage of the unique features offered by voice recognition technology. With the aim of improving the control of spreadsheets through voice recognition, the authors have developed an intelligent navigation system called iVoice, as outlined in Section 4.

An experiment, detailed in Section 5, was conducted to evaluate iVoice against a state of the art voice recognition technology. Section 6 summarizes the results of this experiment and Section 7 concludes this paper.

2 Speech Technology

Spreadsheets can be used to help companies make multi-million euro decisions on a daily basis. Within the financial district of London, spreadsheets have been described as the primary front line tool of analysis[3].

Despite the importance of the decisions being made based on the information contained in these spreadsheets, there is very little done in practice to ensure their quality. Two independant studies [4] [5] have revealed that over 90% of spreadsheets contain errors. These errors can range in severity from simple spelling errors to complicated formula errors.

A range of techniques have been proposed to improve the quality of spreadsheets. Some of these initiatives focus on the importance of development methodologies similar to existing lifecycle processes in place for software development. One such technique is Test Driven Development[6]. This method requires users to write test cases before they develop the spreadsheet. In this way developers are forced to build the solution in small steps and to consider the design at the outset.They will also have a series of tests in place that will ensure the integrity of the spreadsheet as development proceeds.

3 Voice Recognition Technology

Voice recognition technology has been primarily used for creating and modifying text documents[7]. Although vendors of Voice Recognition Technology claim it can be faster than traditional keyboard and mouse input, the author has found no evidence to support this claim. A recent study[8] found that through voice recognition technology, users could quickly achieve input rates of 150 WPM with an accuracy of around 90%.

With the increased interest in voice recognition technology, new applications have emerged across multiple domains. Begel[9] has allowed software developers to create java applications by speaking in a natural way. As the developer speaks, the system, Spoken Java, will generate the java source code.

There are many voice recognition engines available; the best of which is Dragon NaturallySpeaking(DrNS), which boasts an accuracy of 99%. The most basic version, Standard Edition, features support for Microsoft Word and Microsoft Internet Explorer. The Preferred Edition also includes support for Microsoft Excel. Special editions for the legal and medical domains also exist[7].

4 iVoice

In an earlier experiment[2], three experienced spreadsheet users were asked to audit a spreadsheet, seeded with errors, using voice recognition technology. Although none of the participants had any prior experience with voice recognition, they had sufficient experience with spreadsheets to complete the task easily. Their performance was compared to another study[10] in which 13 professional spreadsheet users were asked to audit the same spreadsheet using a keyboard

and mouse. Both studies recorded the behaviour of each participant through logging of cell selection activity via the T-CAT[11] (Time-stamped Cell Activity Tracker) tool. It was found that voice recognition participants found 14% less of the seeded errors despite taking twice as long.

A number of elements, namely editing formulas, entering data and navigation were looked at to try to identify key differences in the behaviour and performance of the groups. The results showed that the voice control users struggled in all of these aspects. Given that navigation is the most fundamental of spreadsheet activities it was decided to explore the development of new technologies that could improve the efficiency of voice-controlled navigation of spreadsheets.

The resulting iVoice system, which integrates with DrNS, provides support for three particular actions, navigation to referenced cells, automatic navigation of a range of similar cells, and navigation directly to the next non-blank cell. Each of the three features is explained and subsequently investigated through an experiment.

4.1 Navigation to Referenced Cells

When looking at the navigational behaviour of all participants in [2], it was observed that upon initial entry to a cell containing a cell reference users would navigate to this cell in order to ensure that the reference was correct. Traditional voice recognition software requires users to navigate to the target worksheet through all intermediary worksheets. iVoice allows users to skip the intermediate worksheets bringing them directly to the remotely referenced cell.

By assigning each referenced cell in a formula a unique colour, users can move directly to the required cell through the command *Jump <colour>*, where *<colour>* is the colour of the desired destination cell. A *Jump Back* command is also provided to enable users to move back to the original cell. It is hypothesised that the time to reach a referenced sheet will, based upon the number of commands, decrease using the iVoice technology for non-adjacent worksheets. The colours and the names used by the system can be displayed and hidden at any time through the *Show Colours* and *Hide Colours* commands.

4.2 Scan Command

The second component of the iVoice system allows users to navigate through a list of semantically similar cells. Semantically similar cells are those whose contents are of a similar structure and purpose[12]. Spreadsheets are in general composed of regions of such cells. It has been observed that users will examine these regions sequentially, spending on average between 0.33 seconds and 1.5 seconds on each cell.

iVoice supports this activity through the provision of a scan command which automatically moves to the next cell in the chosen direction after one second. This delay allows users the chance to review the contents of the cell before moving on. To initiate this command users say *Scan <direction>* where *<direction>* is the direction they wish to scan, be it left, right, up or down. It is hoped that

this feature will reduce the time to perform this task, as it requires a single voice command rather than a series of such commands.

4.3 Jump Blank Cells

The third component of the iVoice system allows users to skip over blank cells. By saying *"Jump <direction>"* where <direction> is the way the user wants to move. The system moves directly to the next non blank cell in that direction. It is felt that this command is more efficient and natural than dictating the associated keyboard shortcut as is currently required in DrNS.

5 Experiment

A quantitative experiment was designed to compare the iVoice system with DrNS. This was followed by a qualitative study, where participants took part in a structured interviewed to establish their view of the iVoice technology.

The quantitative experiment asked six experienced spreadsheet users to highlight, through the *Mark Error* command, as many errors as they could in two spreadsheets through voice recognition technology. This highlighting could be later removed, if required, through the *Unmark Error* command. To randomise the experiment participants were split randomly into two groups of three. Group 1 first audited Spreadsheet 1 using DrNS and then Spreadsheet 2 using iVoice. Group 2 reversed the use of the technologies. The cell selection behaviour of all participants was recorded with the T-CAT tool.

Before the trial commenced participants were asked to configure the voice recognition software. This took approximately ten minutes to complete, after which participants were introduced to the voice-control systems through a navigation exercise which asked them to move about a sample spreadsheet. When participants felt they had mastered the navigation commands, the trial commenced with all participants auditing Spreadsheet 1. No time limit was set for the task allowing participants to finish when they believed they could find no more errors.

The first spreadsheet audited, Spreadsheet 1, was comprised of three worksheets, *Wages*, *Expenses*, and *2007 Department Spending*. The spreadsheet calculates the total expenditure for each of three departments in a company. The *Wages* sheet details all employees wages and which department they belong to. The *Expenses* worksheet is used to detail different expenses to the company and what department these expenses should be assigned to. The final worksheet totals all costs and apportions company-wide expenses to each department based on the number of employees in that department.

When the first spreadsheet had been completed participants were given a break to facilitate the changing of the technology. Participants then received training in the second technology and subsequently audited the second spreadsheet with the second technology. Again, no time limit was set allowing participants to finish when they judged the exercise was complete.

The second spreadsheet, Spreadsheet 2, also contained three worksheets, *Opening Stock*, *Purchases* and *Sales and Profit*. This spreadsheet was used to calculate the profit made on each of 18 products over a given period. The *Opening Stock* worksheet detailed the quantity and value of each of the products at the start of the period. The *Purchases* worksheet detailed the purchases that were made during the period, and the *Sales and Profit* worksheet detailed the sales and closing stock of each product. This worksheet also uses the costs from the *Opening Stock* and *Purchases* worksheets to calculate the profit for the period.

6 Analysis of Results

Before examining the individual features, the overall performance and behaviour of participants using each technology was examined. The results obtained by the T-CAT tool allowed for a detailed analysis of the experiment.

The measures used for performance were spreadsheet coverage and errors found. Evidence has been found to suggest that there is a relationship between the number of cells evaluated and the number of errors found[10]. For the purpose of this experiment coverage was defined as the percentage of the spreadsheet that was actually reviewed by participants where a cell was considered to be reviewed if a participant spent more than 0.3 seconds on it. Only cells that contain numerical data or formula were considered as all other cells could be reviewed without being entered.

Table 1. Cell Coverage

	Spreadsheet 1	Spreadsheet 2
iVoice	87.5	44.6
DrNS	53.6	37.1

Table 1 shows that iVoice users covered a higher percentage of the cells than users of DrNS. In Spreadsheet 1 users who used iVoice covered 87.5% in contrast to the 53.6% covered by those using DrNS. For Spreadsheet 2, using iVoice, participants covered 44.6% of the spreadsheet whereas participants using DrNS covered 37.1%.

Table 2. Time spent auditing spreadsheet in minutes

	Spreadsheet 1	Spreadsheet 2
Group 1	28.4	21.5
Group 2	26.8	20.0

It is important to establish that any difference in performance is not due to a difference in time spent by the groups on the tasks. Table 2 shows the average

time each group spent auditing each spreadsheet. It was found that Group 1 spent on average 90 seconds more on each spreadsheet than those in Group 2, regardless of the technology that was employed. The difference is small and so it was concluded that this effect is insignificant.

Table 3. Overall Performance Per Spreadsheet

	Spreadsheet 1	Spreadsheet 2
iVoice	69.4%	61.1%
DrNS	63.9%	57.4%

Table 3 shows the average percentage of errors that were found by each group. It was found that users of iVoice found between 3% and 6% more than users of DrNS. On Spreadsheet 1 those using iVoice found 69.4% of the errors whereas those using DrNS found 63.9%. For Spreadsheet 2 those using iVoice found 61.1% of the errors whereas those using DrNS found 57.4%. While the sample sizes are insufficient to establish a statistically significant difference these values indicate that iVoice leads to better coverage and performance than Dragon NaturallySpeaking. The results obtained through a more detailed analysis of each of the iVoice features is now presented.

6.1 Navigate to Referenced Cells

The time to navigate to referenced cells is measured from when the participants leave the source cell until they enter the destination cell, omitting the time spent in both the source and destination cell. The average values are quoted only for participants who performed this action three or more times. This feature was used by participants between 1 and 10 times during the experiment.

While using DrNS it was found that users spent on average 4.1 seconds moving from a cell that contains a reference to the referenced worksheet, passing through one intermediate worksheet. Participants needed to spend on average a further 2.7 seconds returning to the original cell. These results indicate that through the use of iVoice users can save approximately seven seconds checking one remote reference, as the iVoice system can bring users directly to such a reference in one command and back to the original cell through a second command. As the number of intermediate worksheets increases it is expected that the savings quoted above would also increase.

6.2 Scan Command

The scan command was evaluated by measuring the average time participants spent in each cell of a scanned region. Each evaluated region contained a minimum of three cells. The first cell in each region was discarded as it is believed users spend longer reviewing this cell and including this time would distort the

results. For the analysis, each participant must have scanned at least three regions. Only one participant failed to scan three such regions. This feature was used between 2 and 25 times by participants during the trial.

Table 4. Overall Performance Per Spreadsheet

	Spreadsheet 1	Spreadsheet 2
iVoice	0.93	0.97
DrNS	2.62	2.92

Table 4 shows average time participants spent on each cell while scanning through a region. It was found that when users used the iVoice function they were able to spend the expected 1 second on each cell in a scanned region. When using DrNS it was found that they spent on average 2.5 to 3.0 seconds on each cell, almost three times as long.

6.3 Jump Blank Cells

In order to evaluate the effectiveness of the Jump Direction command, the time at which users left a cell to the time they entered the next non-blank cell was measured. It was found that participants using Dragon NaturallySpeaking spent approximately 1.3 seconds on average performing this action. With iVoice the equivalent time would be zero as they are brought directly form one cell to the next non-blank cell. Although not all participants used this feature, a number of participants used it extensively, with one participant using it 45 times.

6.4 Discussion With Participants

Upon completion of the quantitative trial, a structured interview was conducted to find out participants opinions of the technologies they had used and their prior level of experience with both spreadsheets and voice recognition technology.

It was found that most of the six participants preferred the iVoice navigation system to Dragon NaturallySpeakings own in-built navigation system. The participants remarked that the iVoice commands made tasks like moving to remote references easier and also mentioned that it was easier to concentrate on auditing the spreadsheet while using iVoice.

7 Conclusions

This paper details a controlled experiment that suggestes the performance of spreadsheet auditors using voice recognition technology can be improved through the use of an intelligent navigation system. The experiment compared the performance of six experienced spreadsheet users using two technologies, Dragon

NaturallySpeaking, a leading voice recognition software and iVoice, an intelligent navigation system. The experiment showed that participants using the iVoice system found a higher percentage of errors than those using Dragon NaturallySpeaking. iVoice simplifies navigation of a spreadsheet through three features; scanning through a range of cells, navigating to references off screen and moving over blank cells.

A number of enhancements were identified through a qualitative evaluation of the technology. At present the scan function stops for one second on each cell. In certain situations this time was found to be unsuitable and by allowing users alter this time the scan function could become more efficient. Other features which support the debugging process were also mentioned. One such feature would enable the scan function to automatically stop on cells which are semantically different from the preceding cells. This may be an indication of an error.

8 Acknoweledgement

This work is supported by the Irish Research Council for Science, Engineering and Technology.

References

1. : The repetitive strain injury association rsi facts and figures[online] available at http://rsi.websitehosting-services.co.uk/facts_&_figures.pdf
2. Flood, D., McDaid, K.: Voice-controlled auditing of spreadsheets. In: Proc. of the Conf. of the European Spreadsheet Risks Interest Group, 2007. (2007)
3. Croll, G.: The importance and criticality of spreadsheets in the city of london. In: Proc. of the Conf. of the European Spreadsheet Risks Interest Group, 2005. (2005)
4. Powell, S.G., Baker, K.R., Lawson, B.: Errors in operational spreadsheets [online] http://mba.tuck.dartmouth.edu/spreadsheet/product_pubs.html (2007)
5. Panko, R.R.: What we know about spreadsheet errors. J. End User Comput. **10**(2) (1998) 15–21
6. McDaid, K., Rust, A., Bishop, B.: Test-driven development: can it work for spreadsheets? In: WEUSE '08: Proceedings of the 4th international workshop on End-user software engineering, New York, NY, USA, ACM (2008) 25–29
7. NaturallySpeaking: http://www.nuance.com/naturallyspeaking (2007)
8. Bailey, G.A.: Achieving speed and accuracy with speech recognition software. In: Americas Conference on Information Systems, 2006. (2006)
9. Begel, A.: Spoken language support for software development. In: VLHCC '04: Proceedings of the 2004 IEEE Symposium on Visual Languages - Human Centric Computing, Washington, DC, USA, IEEE Computer Society (2004) 271–272
10. Bishop, B., McDaid, K.: An empirical study of end-user behaviour in spreadsheet debugging. In: The 3rd Annual Work-In-Progress Meeting Of Psychology of Programming Interest Group, 2007. (2007)
11. Bishop, B., McDaid, K.: Spreadsheet debugging behaviour of expert and novice end-users. In: WEUSE '08: Proceedings of the 4th international workshop on End-user software engineering, New York, NY, USA, ACM (2008) 56–60
12. Clermont, M.: A toolkit for scalable spreadsheet visualization. In: Proc. of the Conference of the European Spreadsheet Risks Interest Group, 2004. (2004)