Facilitating Spreadsheet Manipulation on Mobile Devices Leveraging Speech

Arjun Srinivasan
Georgia Institute of Technology
Atlanta, GA 30332, USA
arjun010@gatech.edu

Bongshin Lee
Microsoft Research
Redmond, WA 98052, USA
bongshin@microsoft.com

John Stasko
Georgia Institute of Technology
Atlanta, GA 30332, USA
stasko@cc.gatech.edu

MobileVis '18 Workshop at CHI 2018, April 21, 2018, Montreal, QC, Canada.
https://mobilevis.github.io/

Abstract

Due to the prevalent use of spreadsheet applications, their user interface on a desktop has evolved significantly over time. However, user interfaces for spreadsheet applications on mobile devices are still in a nascent stage and are typically "ported" versions of their desktop counterparts. In this article, we present a simple tabular data manipulation scenario on mobile devices, leveraging speech input. We describe ways to complement direct manipulation-based input via touch or pen with minimalistic speech-based input. We also briefly discuss an open area for future research, where we could explore the role of speech input for interacting with spreadsheets along the spectrum of command-like utterances to a conversational dialog.

Author Keywords

Multimodal interaction; speech; touch; spreadsheet; visualization; mobile devices.

ACM Classification Keywords

H.5.2 [Information interfaces and presentation (e.g., HCI)]: User Interfaces

Introduction

Millions of people use spreadsheet applications such as Microsoft Excel, Google Sheets, and Apple Numbers to store and manipulate tabular data, as well as to create charts with data. As mobile devices such as smartphones and tablets become practically ubiquitous, it is important to enable effective interaction with spreadsheets on mobile devices. While desktop-based interfaces for spreadsheet applications have evolved significantly over time, mobile-based interfaces for spreadsheet applications are still in a nascent stage and are typically "ported" versions of their desktop counterparts.

Recent work has begun to explore how touch- and pen-based input can be used in innovative ways to enable a more natural and fluid interaction experience when working with spreadsheets on mobile devices. For example, Tableur [16] leverages the notion of "pen writes" [5] in the context of spreadsheets, enabling several data entry and manipulation operations in a naturalistic manner. Another recently presented interaction technique involves bi-manual input combining pen and touch [10] for commonly used spreadsheet tasks within the context of an application like Microsoft Excel. While these techniques are a useful step toward more natural and fluid interaction with tabular datasets, speech interaction can offer additional benefits especially for mobile devices.

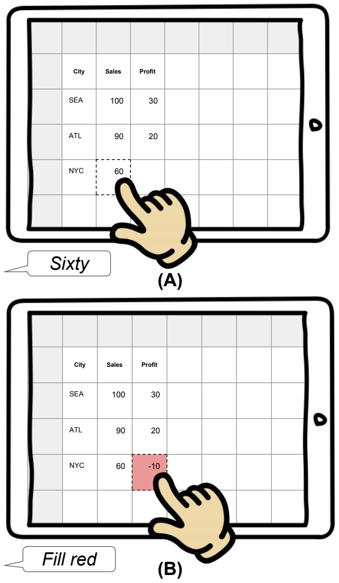
With the advances in speech recognition technology, natural language is gaining increased attention as a plausible way to manage and analyze data (e.g., [4, 1, 13]). Research has shown that multimodal interaction combining touch- or pen-based direct manipulation input with speech can have many advantages, offering a more naturalistic interaction experience (e.g., [8, 9, 14]).

In this article, we describe a simple data manipulation scenario for spreadsheet applications on mobile devices. We consider ways in which even minimalistic speech-based input can be used to complement direct manipulation-based input via touch or pen. The goal of this article is not to propose a definitive or exhaustive set of interactions, nor is it to say that speech-based interaction can replace or perform better than existing techniques. Through exemplary scenarios and interactions, we explore how such multimodal interaction may improve the interaction experience, alleviating some of the common challenges with existing versions of spreadsheet applications on mobile devices.

Spreadsheet Applications on Mobile Devices

Selection, entering data, specifying formulas, and issuing commands are common operations in spreadsheet applications. For example, for simple data entry, you select a target cell and type in a value. For moving a block of cells, you select the block, cut it, and paste it into the target location. These operations are very easy to perform on a desktop that is equipped with a mouse and keyboard, especially if you are familiar with shortcut keys. However, they are very cumbersome on mobile devices for multiple reasons. While text input on mobile devices has improved over the years, it is still not easy to type with a virtual keyboard compared to a physical keyboard. In addition, due to the limited screen space, mobile devices do not show the keyboard all the time, requiring additional interaction to invoke a keyboard. Furthermore, a drag gesture (a simple way of selecting a block in a desktop application) is, by default, mapped to a pan operation because mobile devices cannot afford to lose the area taken by scroll bars. This again requires additional interaction to perform a basic operation such as selecting a block of cells.

To address these issues, researchers have proposed to leverage pen interaction or bi-manual interaction. For example, Tableur incorporates handwriting recognition and offers a set of ink-based gestures to manipulate and organize ink stroke [16]. Once a gesture (e.g., lasso selection) is recognized, Tableur displays a context menu containing possible



City	Sales	Profit
SEA	100	30
ATL	90	20
NYC	60	

City	Sales	Profit
SEA	100	30
ATL	90	20
NYC	60	-10

Figure 1: Using touch and speech simultaneously to enter data and style cells. (A) User points on a cell and says “*Sixty*.” (B) User points on a target cell and says “*Fill red*.”

operations (e.g., Create Table, Toggle Column Header) for the region. Once table is created, users can manually write in values or perform smart filling (i.e., expanding number or date patterns across columns or rows) through a horizontal or vertical line gesture across multiple cells. Another recently presented interaction technique involves bi-manual input combining touch (specifically with thumb) and pen [10] for three different applications including spreadsheets like Microsoft Excel. Specifically, in the context of tablet devices, the non-dominant hand’s thumb input supports the dominant hand’s pen actions. For example, users can trigger a marking menu at the pen location with their thumb, and the select the menu item with the pen. By continuously holding the “Copy” button from the Pen Tools Widget, users can maintain the Copy mode while panning to the target location. However, these techniques rely heavily on context menus which work well when the number of operations are limited, but become challenging to manage as the number of options increase [11]. More importantly, speech input can complement these pen and/or interactions, offering additional benefits especially for mobile devices.

Envisioned Scenario: Talk to Table

Advances in speech recognition technology make it easy to achieve high recognition accuracy for basic grammars with a limited vocabulary. However, developing sophisticated natural language understanding modules that effectively manage issues like ambiguity and context remain an open area for development. Therefore, to make it more practical and feasible with existing technologies, our scenario assumes minimal natural language understanding capabilities. However, while we focus on mobile and tablet devices, these scenarios can be applied to other pen- and/or touch-enabled devices, including wall-size displays such as Microsoft’s Surface Hub.

Add and edit data

With an existing spreadsheet application for mobile devices, to enter or edit data in a cell, users need to either tap the target cell and then the formula bar at the top or double tap the target cell to invoke a keyboard. They then need to type in a data value using a virtual keyboard. Similarly, to perform an edit operation (e.g., cut, copy) on a cell, users first need to tap the target cell followed by another tap to invoke a context menu. To change the style (e.g., fill color) of a cell, users need to follow a series of steps involving the opening of a dialog box, selecting a tab, and specifying a color. A straightforward combination of touch and speech can replace such tedious sequences of interactions with simple operations such as speaking out a data value (e.g., "twenty five"), a command (e.g., "delete" or "cut"), or a command with a desired property (e.g., "fill red") while holding a cell with a finger (Figure 1).

Move or copy cell contents

To select a block of cells in a desktop spreadsheet application, users can simply perform a drag gesture over the block. However, on mobile devices, a drag gesture is typically mapped to a pan operation because these devices (with constrained space) cannot afford to lose the area taken by scroll bars. Therefore, users typically need to use drag handles provided at the corners of a selected block (or cell). Such interaction faces the “fat-finger” problem [12], however, and it also conflicts with the auto-population operation commonly used in desktop versions of spreadsheet applications, and thus can be confusing to users. In addition, existing spreadsheet applications on mobile devices typically require users to interact with context menus [16, 10], making users divert their attention from their action and focus on the context menu options.

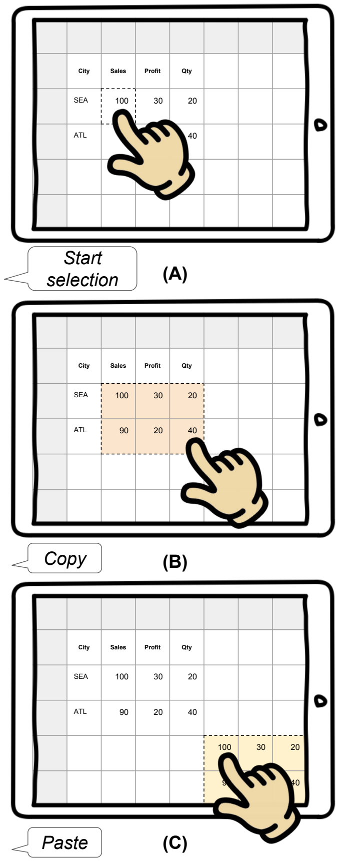


Figure 2: Using touch and speech simultaneously to select a block of cells and copy them. (A) User points on a cell and says “*Start selection*.” (B) User drags finger to the end of the selection block and says “Copy” (C) User points on a target cell and says “Paste.”

We can use speech to complement touch input and alleviate these issues. With a finger holding on a cell, users can issue a “*Start selection*” command. This lets users move in the desired direction without lifting a finger to perform selections similar to a desktop. When they reach the end of the block they wanted, they can issue a command (e.g., "End selection," "cut," or "copy"). For the cases where the target block is large, users can go to the end of the target block using a pan interaction, and then issue a command while holding the cell there. Once the block is copied into the clipboard, users can paste by saying "move" or "paste" while holding a cell with a finger (Figure 2).

Create formulas and simple charts

Creating a formula that derives values using other fields is a commonly used operation in spreadsheet applications. However, as this operation is typically performed by typing formulae within a cell or formula bar, it is particularly challenging to perform on mobile devices. Speech input can be leveraged to let users more easily create formulas in spreadsheet applications. For example, when users want to compute a single value (e.g., average) from a single column, they can simply point at an empty cell in the target column and speak out the function name (without necessarily selecting the column values first as shown in Figure 3A).

Multiple values can also be derived from multiple rows or columns. For example, after selecting two columns, the user points at the header of another column and says “*Add*,” the application can add the sum of the two columns in the pointed-to column (Figure 3B).

Another potential application of speech is to invoke a visualization of the data in a spreadsheet. Speech input can be used to let users directly create the visualization they want without relying on a dialog box or control panel. For example, once a block of cells is selected, users can say “*Line sparkline*” while holding a target cell to create sparklines. Additionally, with a built-in visualization selection module, users could even simply issue a command like “*Visualize*,” the application could automatically create the most suited chart for the user (Figure 4).

Global speech interaction

Some actions (e.g., undo, select all) are applicable to the entire table rather than a specific cell or block. Users can issue spoken commands for global spreadsheet-level actions such as "undo," "clear," and "select all." In addition, they can perform operations on specific parts of the table, for example, saying “*Order [the table] by the second column*.”

Discussion and Future Work

In this article, we have considered speech as a secondary modality to complement direct manipulation-based input for basic spreadsheet manipulation tasks. Examples in this paper have largely assumed the context of a tablet device. However, in smaller devices such as phones or watches, even selecting a cell or a range of cells becomes a challenge and issues such as selection precision and occlusion become even more prominent [12, 15]. Consequently, exploring design alternatives to facilitate touch-based interaction remains an open area for improvement to make possible the multimodal interaction-based examples presented in this article. Furthermore, to limit our scope to immediately applicable and practically realizable use cases with minimal natural language understanding capabilities, we use simple command-like utterances in our examples. However, one of the key advantages of speech input is the high degree of freedom of expression it affords [7]. This freedom of expression can be leveraged to let users easily perform more complex operations that would otherwise require them to perform multiple steps. For example, filtering is an operation that is extensively performed using WIMP

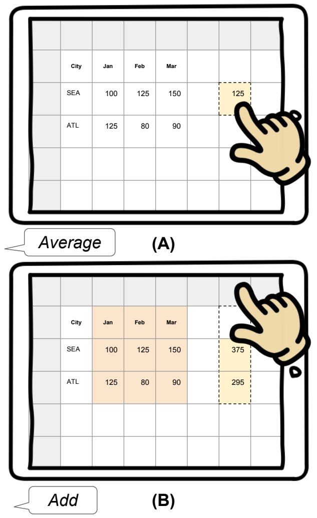


Figure 3: Using touch and speech simultaneously to perform calculations. (A) User points on a cell and says "*Average*" to average corresponding row values. (B) With multiple columns selected, user points at the header of an empty column and says "Add" to create a new calculated column.

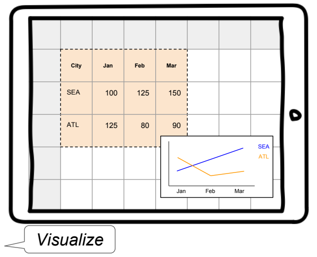


Figure 4: Using speech to visualize a selection block.

interfaces, requiring users to first trigger a filtering mode and then interact with multiple widgets such as drop-downs even for applying simple single value based filters. Leveraging speech, we could enable users to apply filters as a part of their selection query (e.g., "*Select rows with 'profit' greater than 50*" or "*Select rows similar to this one*").

However, the freedom of expression afforded by speech input also implies that the queries themselves can be uttered in several ways (e.g., "*Select all profitable rows*") and that systems would need to be intelligent enough to understand these variations and semantics of the data. Exploring the role of speech along the spectrum of command-like utterances to conversational questions is an exciting open area for future research. As the role of natural language expands from being a complementary modality to an on-demand alternative or even primary input modality, it will be important for designers to think about associated user interface components such as ambiguity widgets [3] and means to overcome other challenges with speech input such as the lack of clear affordances and providing effective feedback about changes made in response to user queries. Further, other issues such as conflicting background noises can be prevalent, hindering accurate recognition when using spoken input particularly in the context of mobile devices [2, 6]. Therefore, providing simple ways to correct recognition errors is another area for future work.

REFERENCES

1. Kedar Dhamdhere, Kevin S McCurley, Ralfi Nahmias, Mukund Sundararajan, and Qiqi Yan. 2017. Analyza: Exploring data with conversation. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces*. ACM, 493–504.
2. Khaled El-Maleh, Ara Samouelian, and Peter Kabal. 1999. Frame level noise classification in mobile environments. In *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*, Vol. 1. IEEE, 237–240.
3. Tong Gao, Mira Dontcheva, Eytan Adar, Zhicheng Liu, and Karrie G Karahalios. 2015. Datatone: Managing ambiguity in natural language interfaces for data visualization. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, 489–500.
4. Sumit Gulwani and Mark Marron. 2014. Nlyze: Interactive programming by natural language for spreadsheet data analysis and manipulation. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*. ACM, 803–814.
5. Ken Hinckley, Koji Yatani, Michel Pahud, Nicole Coddington, Jenny Rodenhouse, Andy Wilson, Hrvoje Benko, and Bill Buxton. 2010. Pen+ touch= new tools. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*. ACM, 27–36.
6. Hans-Günter Hirsch and David Pearce. 2000. The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions. In *ASR2000-Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop (ITRW)*.
7. Bongshin Lee, Petra Isenberg, Nathalie Henry Riche, and Sheelagh Carpendale. 2012. Beyond mouse and keyboard: Expanding design considerations for information visualization interactions. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (2012), 2689–2698.
8. Sharon Oviatt. 1997. Multimodal interactive maps: Designing for human performance. *Human-computer interaction* 12, 1 (1997), 93–129.

9. Sharon Oviatt, Rachel Coulston, and Rebecca Lunsford. 2004. When do we interact multimodally?: cognitive load and multimodal communication patterns. In *Proceedings of the 6th international conference on Multimodal interfaces*. ACM, 129–136.

10. Ken Pfeuffer, Ken Hinckley, Michel Pahud, and Bill Buxton. 2017. Thumb+ Pen Interaction on Tablets.. In *CHI*. 3254–3266.

11. Ben Shneiderman and Pattie Maes. 1997. Direct manipulation vs. interface agents. *interactions* 4, 6 (1997), 42–61.

12. Katie A Siek, Yvonne Rogers, and Kay H Connelly. 2005. Fat finger worries: how older and younger users physically interact with PDAs. In *IFIP Conference on Human-Computer Interaction*. Springer, 267–280.

13. Arjun Srinivasan and John Stasko. 2017. Natural language interfaces for data analysis with visualization: Considering what has and could be asked. In *Proceedings of EuroVis*, Vol. 17. 55–59.

14. Arjun Srinivasan and John Stasko. 2018. Orko: Facilitating Multimodal Interaction for Visual Exploration and Analysis of Networks. *IEEE transactions on visualization and computer graphics* 24, 1 (2018), 511–521.

15. Daniel Vogel and Patrick Baudisch. 2007. Shift: a technique for operating pen-based interfaces using touch. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 657–666.

16. Emanuel Zgraggen, Robert Zeleznik, and Philipp Eichmann. 2016. Tableur: Handwritten spreadsheets. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, 2362–2368.