MaxSorting: A New Card Sorting Tool

Yun-Ju Becker Lee
Department of Industrial Engineering & Management
National Taipei University of Technology (Taipei Tech), Taipei, Taiwan
Tel: (+886) 2- 2771-2171 ext. 2385, Email: bws723@hotmail.com

Sheau-Farn Max Liang
Department of Industrial Engineering & Management
National Taipei University of Technology (Taipei Tech), Taipei, Taiwan
Tel: (+886) 2- 2771-2171 ext. 2345, Email: maxliang@ntut.edu.tw

Abstract. Card sorting is an efficient technique to measure human mental models and has been applied in various research areas. However, there is room for improvement for current card sorting software applications due to a lack of flexibility in data analysis and an inconvenient drag-and-drop user interface found in most of these applications. A new application, termed MaxSorting, has therefore been developed to include a series of analysis methods and to feature a point-and-click user interface. Results of a usability test against current card sorting software tools showed that MaxSorting was a more effective and efficient tool in terms of its fewer numbers of errors and less completion times. In summary, with its capability to calculate and report eight different similarity coefficients among sorted cards and edit distances among participants, MaxSorting is a useful and usable card sorting tool for researchers to understand human mental models and to design information architectures.

Keywords: Card sorting, Mental model, Similarity coefficient, Edit distance

1. INTRODUCTION

MaxSorting, a new card sorting tool, is introduced in this paper, and its usability was evaluated against current card sorting tools. The background of card sorting, the motivation and purpose of this paper, and the design of MaxSorting are presented in the following sections.

1.1 Background of Card Sorting

Card sorting method has been applied in various fields for years to measure human mental models. It is a fast, systematic, and easy-to-use technology (Mohammed et al., 2000). Originated in Kelly's (1955) Personal Construct Theory, card sorting is a tool for people to group different items in accordance with their thought. Each card can represent an object, a thing, or a concept. Through the sorting process, the knowledge structure or mental model about these cards can be derived and understood. Card sorting technique has been applied in training research to find differences in knowledge structures between more experienced trainees and less experienced trainees (Smith, Jentsch et al, 2001; Liang, 2008). Liang et al. (2009) applied card sorting tool to analyze nuclear power plant operators' mental models. In website related research, card sorting tool has been applied in evaluating a website usability and enhancing the effectiveness and efficiency of product search (Rau and Liang, 2003; Liang and Yang, 2008).

Card sorting was usually conducted through physical cards, but current card sorting tools are available to be used via personal computers, such as the IBM EZCalc/USort applications (Dong et al, 2001), or via websites, such as WebSort and OptimalSort. Figure 1 shows a typical user interface of these card sorting applications with two main areas. The left area is for "the list of card items"; the right area is "grouping area." The operation procedures on the user interface are that users first drag a card from "the list of card items" into the "grouping area," then they arrange similar card items together as a group. Finally, they type group names at "group name of inputting" and click finish button.

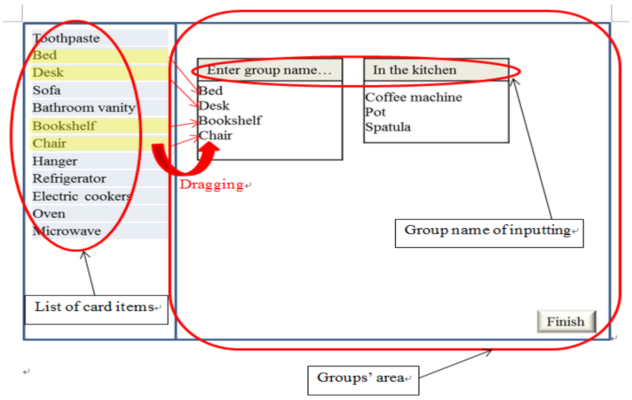

Figure 1: Typical user interface of current card sorting applications.

1.2 Motivation and Purpose

Card sorting applications are useful tools for researchers. However, one drawback of current card sorting applications is that their analysis algorithms are based on only one of many similarity coefficients. Similarity coefficients are used to measure how similar any two card items are. Its value ranges from zero (0) to one (1). One (1) means the highest degree of similarity, whereas zero (0) indicates the lowest degree. There are many formulas to calculate the similarity coefficients. Meyer et al. (2004) has presented the eight most recognized formulas: Jaccard (1901), Dice (1945), Anderberg (1973), Ochiai (1957), Simple Matching (Sokal and Michener, 1958), Rogers-Tanimoto (1960), Ochiai II (Ochiai, 1957), and Russell-Rao (1940). Furthermore, Liang and Tzeng (2012) have found that Jaccard, Dice, Anderberg, and Ochiai were more appropriate than Simple Matching, Rogers-Tanimoto, and Russell-Rao to measure human mental model. Therefore, these appropriate similarity coefficients should be available in any card sorting tool to provide more choices of analysis methods for researchers.

On the other hand, user interfaces (UIs) of card sorting applications are also important to users in card sort tasks. However, the drag-and-drop UI style in current card sorting applications might not be a suitable design. Many researchers examined differences in speed and accuracy between mouse interaction styles of drag-and-drop and point-and-click (Boritz et al., 1991; Gillan et al., 1990; MacKenzie, 1992a; 1992b; 1991). MacKenzie et al. (1991) found that the clicking method on various tasks was faster and less errors than the dragging method. Even study on children, the result was the same as on adults (Inkpen, 2001). Therefore, a new sorting application, MaxSorting, with the flexibility to select many similarity coefficients and the point-and-click UI style, was developed.

1.3 Design of Maxsorting

MaxSorting provides eight most recognized formulas of similarity coefficients (Meyer et al., 2004) compared to only one formula in current card sorting applications. Furthermore, MaxSorting also can measure the difference between two structures, such as two strings (Wager and Fisher, 1974) or trees (Tai, 1979) by applying the calculation of Edit Distances (Papadimitriou and Steglitz, 1982). For example, Liang (2008) measured the difference of knowledge structures between an expert group and a novice group by comparing the edit distances within each group. This feature provides an extra useful tool for the analysis.

For the user interface design, MaxSorting applied point-and-click instead of drag-and-drop style. Card items were arranged close to each other to reduce the resources of scanning and attention (Wickens and Andre, 1990). Figure 2 shows that the main user interface of MaxSorting was divided into three areas from top to bottom: “group name of inputting,” “color selection,” and “the list of card items.” The operation procedures of MaxSorting were as follows: First, users selected a color template from the “color selection” area. Second, on “the list of card items” area, they click the card items that to be grouped and these items would automatically be labeled in the same color. Also, the “group name of inputting” area would appear the corresponding color for each group. Finally, they typed group names at “group name of inputting” area and click the finish button. In “the list of card items” area, card items with the same colors were automatically arranged together to avoid scattered display. Norman (1988) suggested that user interface should add some restrictions to prevent users making mistakes. Compared to the drag-and-drop style, the point-and-click style in MaxSorting could limit users to drag card items to any place on the screen and reduce the difficulty in finding or arranging them later. Furthermore, users could not upload sorting data without entering group names or grouping all the card items. Therefore, we anticipate that users should complete a card sorting task more efficient and make fewer mistakes with MaxSorting than with current card sorting applications.

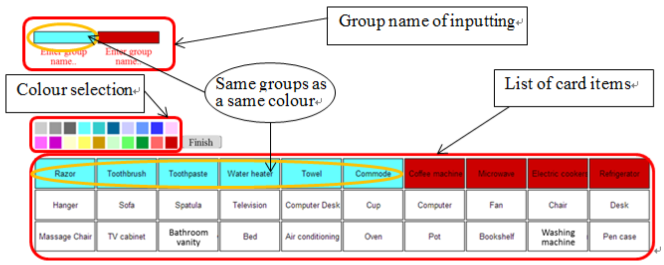

Figure 2: User interface of Maxsorting

2. EVALUATION

The usability of MaxSorting was evaluated against two card sorting applications. Data about task completion time, number of errors, and user satisfaction were collected for analysis.

2.1 Participants

Thirty-six participants, 18 males and 18 females, with age between 23 and 27 years old (M = 24.3) were recruited in this study. A within-subject experiment was designed. That is, participants performed the tasks with all the three card sorting applications. To avoid potential carryover effect due to the performing sequences of the three card sorting applications, participants were divided into six groups. Each group, with 3 males and 3 females, was assigned to one of six possible performing sequences to counterbalance this effect. There were no significant differences in age and gender among these groups, and all the participants reported using internet on computers for more than five years.

2.2 Materials

Two popular online card sorting applications, WebSort and OptimalSort, were chosen to compare with MaxSorting. Both of their user interface layouts and styles were similar to the description in Figure 1. A computer running Windows XP and a 22-inch, 1680x1050 resolution monitor were used to run the experiment. A list of 30 common daily products was selected to be the sort items.

2.3 Data Collection

In the beginning, all the participants were asked to sign a consent form and complete a background questionnaire regarding their experience in computers. Then, they practiced card sorting by using physical cards representing the 30 common daily products so that all the participants could build their mental models on the 30 products before they performed sorting tasks on the computer. In this way we could assume that task completion time on the computer was mainly based on the time spent on interacting with user interface but not the time to think which item should be in which group.

Next, participants were asked to complete four actions on three card sorting applications: (a) sort the 30 card items into groups by their established mental models, (b) change three card items' groups to another groups, (c) give each group a name, and (d) upload the data and finish the sorting task. Finally, participants were asked to fill out the System Usability Scale (SUS) (Brooke, 1996) questionnaire.

2.4 Data Analysis

While the type of card sorting applications was the independent variable with three levels, number of errors, task completion time, and result of the SUS questionnaire were dependent variables. Three types of errors were identified to code errors. They were sorting error, discontinuing error, and non-uploading error. The sorting error was a type of commission error which means doing something incorrectly, and discontinuing error and non-uploading error were a type of omission error which is forgetting to do something (Swain and Guttman, 1983). The sorting error defined as the error due to inappropriate sorting actions. Discontinuing error was the case that participants could not continue to perform the tasks, such as not knowing how to combine two groups to a larger group or to cancel one group directly. As the result, they had to ask the experimenter for help. Non-uploading error represented the error that participants did not upload the data by forgetting to press the confirm button at the end of sorting procedures.

3. RESULTS

Results of the statistical analysis on number of errors, task completion time, and result of the SUS questionnaire among the WebSort, MaxSorting and OptimalSort are presented in the following sections.

3.1 Number of Errors

Total 46 errors had been identified through the 108 (36 participants x 3 applications) trials. By using WebSort, there were nine sorting errors, 10 discontinuing errors, and four non-uploading errors. By using MaxSorting, no sorting error and non-uploading error were found, but there were three discontinuing errors. Using OptimalSort caused 10 sorting errors, nine discontinuing errors, and two non-uploading errors. These errors transformed into the error rates by each card sorting application shown in Figure 3.

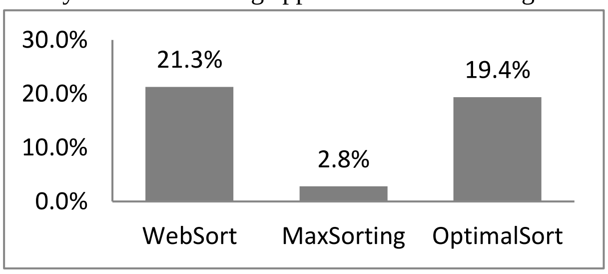

Figure 3: Error Rates by the Three Sorting Applications

Significant differences in number of errors among the three card sorting applications were found ($p<.001$) by the Pearson Chi-square test, and MaxSorting caused the fewest errors.

3.2 Task Completion Time

As shown in Figure 4, the mean task completion times for using WebSort, MaxSorting, and OptimalSort were 187.3, 163.0, and 196.9 seconds, respectively. Significant differences in task completion time across the three applications were found ($F_{2,57} = 3.81$, $p=0.028$) through the one-way ANOVA test. Post-hoc comparisons revealed that task completion time was significantly faster by using MaxSorting than by using WebSort or OptimalSort.

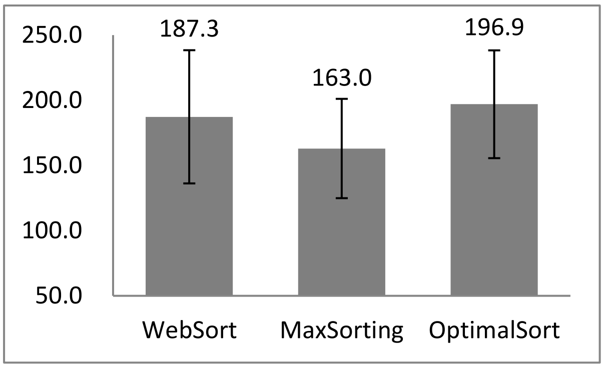

Figure 4: Mean Task Completion Times (in second)

3.3 Subjective Rating

An independent T-Test on the results of SUS questionnaire revealed that the ratings on the three card sorting applications were all significantly above the average score of 50 (WebSort $t_{35} = 4.65$, $p<0.05$; MaxSorting $t_{35} = 6.37$, $p<0.05$; OptimalSort $t_{35} = 6.83$, $p<0.05$), but there were no significant differences in SUS ratings among the three applications ($F_{2,105} = 0.99$, $p=0.37$). This indicated that all the three applications were acceptable by the participants.

4. DISCUSSION AND CONCLUSION

Compared to many sorting errors by using WebSort and OptimalSort, there was no sorting error when MaxSorting was used. The reason might be that the drag-and-drop user interfaces in current card sorting applications provided users too much freedom to drag card items and drop them anywhere without restriction. Some participants did not know where to put the items to form a group. In contrast, the point-and-click user interfaces in MaxSorting place restriction on move any sort item so that the sorting error could be avoided.

Another advantage of pick-and-click over drag-and-drop style is about the discontinuing error. When users' mental models about operation actions differ from the default settings, discontinuing error would occur. Results showed that the number of discontinuing error was fewer by using MaxSorting than by using WebSort or OptimalSort. A possible explanation is that the card items displayed in MaxSorting were fixed buttons which could only be pointed and clicked. The only job for participants to do is to change the colors of card items. However, the card items displayed in WebSort and OptimalSort could be dragged and dropped to anywhere. Participants sometimes forgot where the card items they dropped to and were difficult to find specific card items from the scatter display.

Finally, participants made non-uploading error because the confirm button were not obvious to see so they did not click the button to finish the task. The confirm button in MaxSorting was more obvious than the designs in WebSort and OptimalSort to reduce this type of error.

The difference between the point-and-click and drag-and-drop user interface styles appears to also affect task completion time. This study found that participants took less time to group items with MaxSorting than with WebSort or OptimalSort. It seems that participants had to perform more actions with WebSort or OptimalSort than with MaxSorting. For example, in the drag-and-drop style, participants had to press mouse button, drag a card item to the destination, and release mouse button. On the other hand, with MaxSorting, participants just needed to press mouse button and release. It is obvious that the dragging actions take extra time to complete the task.

Another factor to affect task completion time could be the design of sorting process. In MaxSorting, participants sorted card items in the same group by assigning the same color to the items. System then would automatically arrange sorted groups according to their colors. Conversely, in WebSort or OptimalSort, participants sorted card items one at a time. If there were more than one group, participants had to check all of the groups and decide in which group the current item should be. This takes more time than the sorting process in MaxSorting.

In conclusion, MaxSorting is more effective and efficient than current card sorting applications. With its flexible and comprehensive calculation ability and ease-of-use user interfaces, we anticipate that MaxSorting will be a useful and powerful card sorting tool to assist researchers in dealing with the measurement and comparison of human mental models and knowledge structures.

REFERENCES

Anderberg, M. R. (1973) Cluster analysis for applications. New York:Academic Press.

Brooke, J. (1996) SUS: A quick and dirty usability scale. In P. Jordan, B. Thomas, B. Weerdmeester, and I. L. McClelland, Eds. *Usability evaluation in industry*, 189-94.

Boritz, J., Booth, K. S., and Cowan, W. B. (1991) Fitts' law studies of directional mouse movement. *In Proceedings of the Conference on Graphics Interface '91 (Calgary, Alberta, June 3–7)*, W. A. Davis and B. Wyvill, Chairs. Morgan Kaufmann Publishers Inc., San Francisco, CA, 216–223.

Chaparro B.S., Hinkle V.D., Riley S.K. (2008) The usability of computerized card sorting: a comparison of three applications by researchers and end users. *Journal of Usability Studies*, pp. 31-48.

Dong J., Martin S., and Waldo P. (2001) A user input and analysis tool for information architecture. *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, 23-24.

Dice, L. R. (1945) Measures of the amount of ecologic association between species. *Ecology*, 26, 297-302.

Gillan, D. J., Holden, K., Adam, S., Rudisill, M., AND Magee, L. (1990) How does Fitts' law fit pointing and dragging?. *In Proceedings of the ACM Conference on Human Factors in Computing Systems: Empowering People (CHI '90, Seattle, WA, Apr. 1–5)*, J. C. Chew and J. Whiteside, Eds. ACM Press, New York, NY, 227–234.

Inkpen, K. M. (2001) Drag-and-drop versus point-and-click mouse interaction styles for children. *ACM Transactions on Computer-Human Interaction*, 8 (1), March, 1–33.

Jaccard, P. (1901) Etude comparative de la distribution florale dans une portion des Alpes et du Jura. *Bulletin de la society voudoise des sciences naturelles*, **37**, 547-579.

Kelly, G. A. (1955) The psychology of personal constructs. W.W. Norton, New York.

Liang S. -F. M. (2008) Measuring the convergence and accuracy of trainees' knowledge structures for TFT-LCD visual defect categorization. Int. J. Ind. *Ergonom*., **38**, 307–313.

Liang S. -F. M., and Yang C. –T. (2008) Applying card sorting method to cluster products on shopping websites: A case study, *Proceedings of the 2nd. International Conference on Applied Human Factors and Ergonomics*, 8 pages (CD-ROM).

Liang S. -F. M., Lu C. –H., Hwang S. –L., Chen P. –Y., and Chuang C. –F. (2009) Measuring the consistency between the conceptual model of a nuclear power system and operator mental models, *Proceedings of the 6th. American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Controls, and Human Machine Interface Technology*, 10 pages (CD-ROM).

Liang S. -F. M., and Tzeng L. –W. (2012) Assessing suitability of similarity coefficients in measuring human mental models, *Southeast Asian Network of Ergonomics Societies Conference* (SEANES), 1-5.

Mackenzie, I. S. (1992a) Fitts' law as a research and design tool in human-computer interaction. *Human-Comput. Interact*. **7** (1), 91–139.

Mackenzie, I. S. (1992b) Movement time prediction in human-computer interfaces. *In Proceedings of the Conference on Graphics Interface '92 (Vancouver, BC, Canada, May 11–15), K. S. Booth and A. Fournier, Eds. Morgan Kaufmann Publishers Inc.*, San Francisco, CA, 140–150.

Mackenzie, I. S., Sellen, A., AND Buxton, W. A. S. (1991) A comparison of input devices in elemental pointing and dragging tasks. *In Proceedings of the Conference on Human Factors in Computing Systems: Reaching through Technology (*CHI '91, New Orleans, LA, Apr. 27–May 2), S. P. Robertson, G. M. Olson, and J. S. Olson, Eds. ACM Press, New York, NY, 161–166.

Meyer A. S., F. Garcia A. A., Souza A. P., and Jr. Souza C. L. (2004) Comparison of similarity coefficients used for cluster analysis with dominant markers in maize (Zea mays L). *Genet. Mol. Biol*. **27** (1), 83–91.

Mohammed, S., Klimoski, R., and Rentsch, J.R. (2000) The measurement of team mental model: we have no shared schema. *Organizational Research Methods*, **3** (2), 123–165.

Norman, D. (1988) The psychology of everyday things. *Basic books*, New York.

Ochiai, A. (1957) Zoogeographic studies on the soleoid fishes found in Japan and its neighboring regions. *Bulletin of the Japanese Society of Fish Science*, **22**, 526-530.

Papadimitriou, C. H. and Steiglitz, K. (1982) Combinatorial optimization: Algorithms and complexity. Printice-Hall, New Jersey.Rogers J. S., and Tanimoto, T. T. (1960) A computer program for classing plants. *Science*, *132*, 1115-1118.

Russell P. F., and Rao, T. R. (1940) On habitat and association of species of anophelinae larvae in south-eastern Madras. *Journal Malaria Institute India*, **3**, 153-178.

Rau, P. -L. P., and Liang, S. -F. M. (2003) Internationalization and localization: evaluating and testing a web site for Asian users. *Ergonomics*, **46** (1-3), 255–270.

Smith-Jentsch, K. A., Campbell, G. E., Milanovich, D. M., and Reynolds, A. M., (2001) Measuring teamwork mental models to support training needs assessment, development, and evaluation: two empirical studies. *Journal of Organizational Behavior*, **22** (2), 179–194 machinery.

Sokal R. R., and Michener, C. D. (1958) A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, **38**, 1409-1438.

Swain, A., and Guttman, H. (1983) *Handbook of Human Reliability Analysis with Emphasis on Nuclear Power Plants Applications* (NUREG/CR-1278). Washington, DC, Nuclear Regulatory Commission.

Tai, K.–C. (1979) The tree-to-tree correlation problem. *Journal of the Association for Computing Machinery* **26** (3), 422–433.

Wager, R.A., Fisher, M.J. (1974) The string-to-string correlation problem. *Journal of the Association for Computing Machinery* 21 (1), 168–173.

Wang, M. J., and Drury, C. G. (1989) A method of evaluating inspector's performance differences and job requirements. *Applied Ergonomics*, **20** (3), 181-190.

Wickens C. D., and Andre A. D. (1990) Proximity Compatibility and information display: effects of color, space, and objectness on information integration, *Human Factors*, 61-77.