An Adaptive Keyboard with Personalized Language-based Features

Siska Fitrianie and Leon J.M. Rothkrantz

Man-Machine Interaction, Delft University of Technology,
Mekelweg 4, 2628CD Delft, the Netherlands
{s.fitrianie,l.j.m.rothkrantz}@ewi.tudelft.nl

Abstract. Our research is about an adaptive keyboard, which autonomously adjusts its predictive features and key displays to current user input. We used personalized word prediction to improve the performance of such a system. Prediction using common English dictionary (represented by the British National Corpus) is compared with prediction using personal data, such as personal documents, chat logs, and personal emails. A user study was also conducted to gather requirements for a new keyboard design. Based on these studies, we develop a personalized and adaptive on-screen keyboard for both single-handed and zero-handed users. It combines tapping-based and motion-based text input with language-based acceleration techniques, including personalized and adaptive task-based dictionary, frequent character prompting, word completion, and grammar checker with suffix completion.

Key words: Adaptive interface, personalized system, human-user-interaction, word prediction and completion, on-screen keyboard

1 Introduction

With embedded technology and connectivity, mobile devices and wearable computers are progressively smaller and more powerful. Such devices offer users freeing one or both hands for mobile activity demands. Alternative input devices have been developed to support operating the devices, such as single-handed (e.g. joystick, pen and touchscreen, trackball, and mouse) and zero-handed input devices (e.g. head-mouse or gaze-tracker). These input devices are also used to assist disabled people for interacting with computers [17]. This type of users may have lost the use of one or both hands. Some of them rely on computers to bridge communication with others.

Despite of these developments, text input is still a bottle-neck [15]. Improvement in the input method performance is still highly desired. While speech input [12] and handwriting recognition technology [3] continue to improve, pointing-based character entry is still the most popular to use. With pointing-based keyboards (on-screen), inserting character is strictly sequential. The distance to travel from one key to the next [14] and time for distinguishing an individual character from the group [5] have major effects on the text entry performance.

Familiarity with the location of characters [10] and visual cues to draw attention to the next most probable character(s) in a currently typed word [11][22] can facilitate the performance. The predictive feature can also suggest word completion beginning with the characters that have been inputted so far. The user can select the suggested word or continue to input until the desired word appears.

In this paper, we present our studies on a comparison of common English and personalized dictionary for improving the word prediction of an adaptive keyboard. We also gathered user requirements for developing such a system. The results are used to develop a new on-screen keyboard that can collect knowledge about user linguistics compositions and use the knowledge to alter its future interaction. It has an n-gram based word-level prediction based on the user's personal way of formulating language, the user's task and the English syntax.

The paper is structured as follows. In section 2, we present related work. We continue with presenting our studies in section 3 and 4, respectively. Then, our keyboard model and its word prediction are described in section 5. Finally, we conclude the work in section 6.

2 Related Work

Some alternative keyboard layouts (other than QWERTY) with movement minimizing were developed recently, such as (1) tapping-based (clicking-based) entries, e.g. Fitaly [16], (2) motion-based (gesture-based) entries, e.g. Cirrin [13], and (3) hybrid-based entries, e.g. ATOMIC [22]. Cirrin (Fig. 2(a)) arranges the characters inside the perimeter of an annulus. The most commonly used digrams are nearest to each other, therefore distances traveled between characters are shorter than QWERTY. However, since there is not any predictive feature, a user must attend to the interface when entering text.

Some input techniques have been developed adaptive and with predictive features. Dasher uses prediction by partial matching, in which a set of previous symbols in the uncompressed symbol stream is used to predict the next symbol in the stream [19]. It employs continuous input by dynamically arranging characters in multiple columns positioning the next most likely character near the user's cursor pointer in boxes sized according to their relative probabilities. An icon-based keyboard developed by [7] rearranges most relevant icons to the user's input context (on or) around the center with different icon sizes according to their relative probabilities.

Word prediction/completion can improve entry performance but searching through its word list is considered as tedious and disruptive [2]. Moreover, since statistical models are considered weak in capturing long-distance co-occurrence relations between words, small amount improvement on word prediction can be achieved by using syntactic information in the prediction, such as part-of-speech n-gram information [8]. In contrast, Windmill uses a parsing algorithm for excluding implausible or ungrammatical words from its word prediction's input [20]. Most of these grammar checkers employ a part-of-speech tagger and a set of pattern matching rules [9].

3 Experiment: Common or Personalized Dictionary

The purpose of this experiment was to find out: (1) can a word prediction be improved by using personalized dictionary? and (2) which and how personal data should be used? We collected four datasets from: (1) common English from British National Corpus (BNC - 166261 words) [1], (2) 4.4 MB personal documents in the multimodal communication field (19121 words), such as documents, spreadsheets, and schedulers, (3) 7.2 MB corporate e-mails (13046 words) from the Enron Co., an energy company in Texas [4], and (4) 4.2 MB chat-logs (15432 words) that contain discussions about life aftertime, science and aliens [21].

As the first step, we compared the coverage of the BNC to words and bigrams in personal datasets. 5500 most frequent words (at least 20 times) of the personal datasets were selected. Table 1 shows that in average 87% of words in personal datasets and about 74% of the union of all personal datasets are covered by the BNC. Most words that are not covered by the BNC are names and specific terms, e.g. "xtag", "wordnet", and "website" in the personal documents, "teleconference" and "unsubscribe" in the e-mails, and "lol" (laugh out loud), "yup" (OK), and emoticons in the chat-logs. We selected bigrams containing words that were covered by the BNC. Table 1 shows that although all words are covered, their combinations may not, which are terminologies in a specific domain. For example, (1) in the personal documents: "input fusion", "modality conversion" and "multimodal dialogue" in the field of multimodal system and "shallow parsing", "pattern matching" and "speech recognition" in the field of NLP - they are considered as high frequent bigrams (at least 29 times), (2) in the e-mails: "employee meeting", "management report" and "retirement plans" in corporate domain and "intended recipient", "conference call" and "video connection" in communication field, and (3) in the chat-logs: "immune system", "orbital path", "aftertime life" and "underground shelters".

Table 1. The coverage of BNC toward the personal datasets

	#Words	BNC Cov. (166261 words)	$A \cup B \cup C$ Words Cov.	#Bigrams	BNC Cov. (726000 bigrams)	$A \cup B \cup C$ Bigrams Cov.
A:Personal Docs	5500	4982 (90%)	49%	54829	33994 (62%)	56%
B:E-mails	5500	4740 (86%)	49%	10505	7016 (83%)	11%
C:Chat Logs	5500	4754 (86%)	49%	36801	29809 (81%)	37%
$A \cap B \cap C$	1685	1674 (99%)	15%	2426	2348 (96%)	2.4%
$A \cup B \cup C$	11168	9579 (85%)		89275	68742 (77%)	

These experimental results show that user personal word usage has a strong correlation with the user's task context. The coverage of the BNC to the intersection of the personal datasets is quite high. However, among the personal

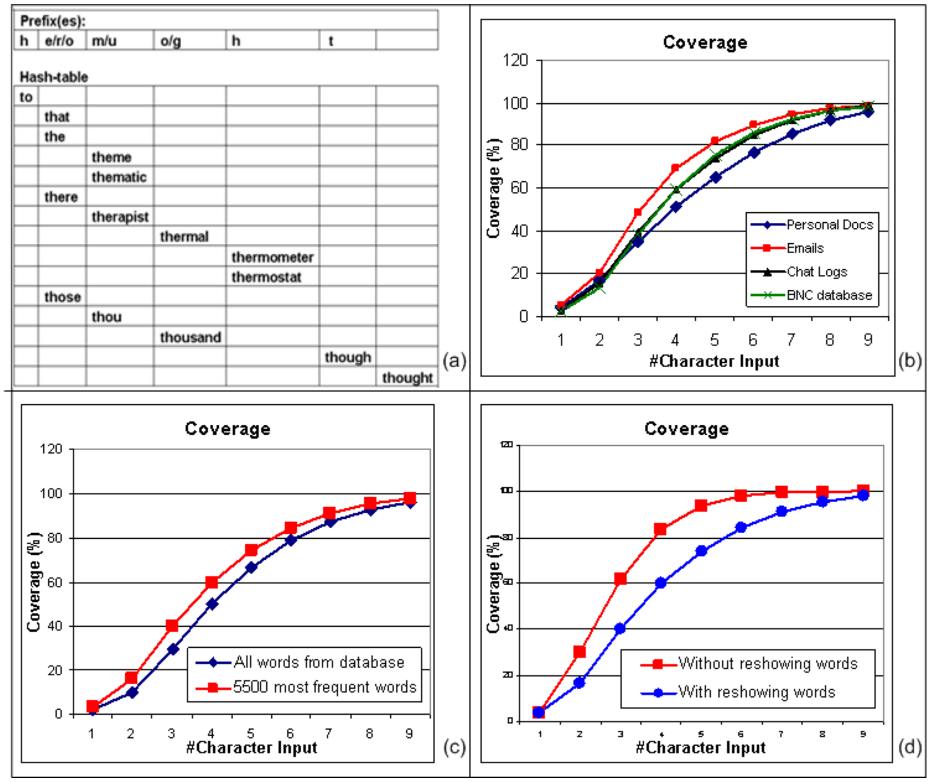

Fig. 1. (a) A part of a hierarchical hash-table for the first character "t" (schematic view - read from left to right), (b) coverage of 5500 most frequent words, (c) average coverage of all words and 5500 most frequent words, and (d) average coverage of the 5500 most frequent words with reshowing words and without reshowing words

datasets shares only a small amount. The reason could be that the datasets were from a specific context and/or not from the same source.

In second step, we simulated word completion without any statistical model using hash-tables. Fig. 1(a) shows user character entries to serve as a prefix before a completion of a word. Different columns show that some characters are necessary for completing the word, e.g. for "thermometer" needs "t", "h", "e", "r", "m", and "o" to distinguish it from "thermal". Fig. 1(b) shows that on average 3.6% of the cases, a user is able to select an intended word in just one entry. Almost similar coverage in all datasets occurs for every prefix. Fig. 1(c) shows degradation of the performance of the completion if a complete set of the datasets is used because of the inclusion of lower frequency words. Fig. 1(d) shows if the completion is not reshowing the same word completions once these words have been shown for a given word being entered. For example, when "ther" is written, "thermal" is one possible completion. If "m" is inputted next, a better option is to show a different word completion, e.g. "thermometer".

4 User Study: Requirements

Dasher is an adaptive on-screen keyboard system for both single and zero-handed users. As reported in [19], although the creators claimed that Dasher needs short training time, its users' text entry rate is less than QWERTY layout's. Moreover, typical writing errors were spelling and syntax errors, which were reduced after some training. There was not any report about its user's satisfaction.

We conducted an interview with a Dasher user, to gather more requirements for developing a new adaptive on-screen keyboard. Our participant is a computer science student. He suffers from cerebral palsy, which impairs physical movement and limits speech. To enable him to communicate he uses a computer device. He has used Dasher for two years with a head-tracker device. The only reason is because Dasher is motion-based text entry. Although our participant claimed that Dasher is easy to use, but he needed some time to learn it. He always uses Dasher's word prediction. The boxes sizes and color contrasts are very important for him as visual cues for next character selections. However, because the character arrangement constantly changes, Dasher demands its user's visual attention to dynamically react to the changing layout. This makes him dizzy after some time. Moreover, it is not always easy to correct errors, since Dasher's interface does not provide fast error recovery button/menu. The current implementation helps him in writing text and documents, but is less suitable for writing in specific context like daily talks, e-mailing, chatting, emergency noting and programming. It is desirable to have such text entry device that works in specific domains with a personalized vocabulary.

5 A Personalized-Adaptive On-Screen Keyboard

Fig. 2(b) shows our developed on-screen keyboard. We adopt the design of Cirrin ([13] - Fig. 2(a)) by displaying all characters in a circular way. Therefore, the interface can gives visual cues, such as different key sizes and color contrasts, for frequently used characters according to their relative probabilities without changing the character layout. Besides this cue, our developed keyboard offers a fast input, less visually demanding and fast error recovery by four ways: (1) the most likely completions of the partially typed word (both user's input and its completion shown in the middle of circle), (2) combining both tapping- and motion-based input (tapping is easier for novice users - [22]), (3) adding space and backspace into the circle for fast error recovery, and (4) each suggested word is shown once after it is rejected by selecting the next character for better language coverage. An additional matrix 6 x 5 is placed on the right side of the circle for numbers, shift, return, control, period, punctuations and comma.

When entering a word, the user may begin with the tapping mode and continue with the motion mode, or vice versa, or only one of them. New selections will be appended to the previous selections. In the motion mode, dragging starts and ends in the middle of the circle. When the user stops dragging, a space will be added at the end of the word. When a space is selected, the input will be

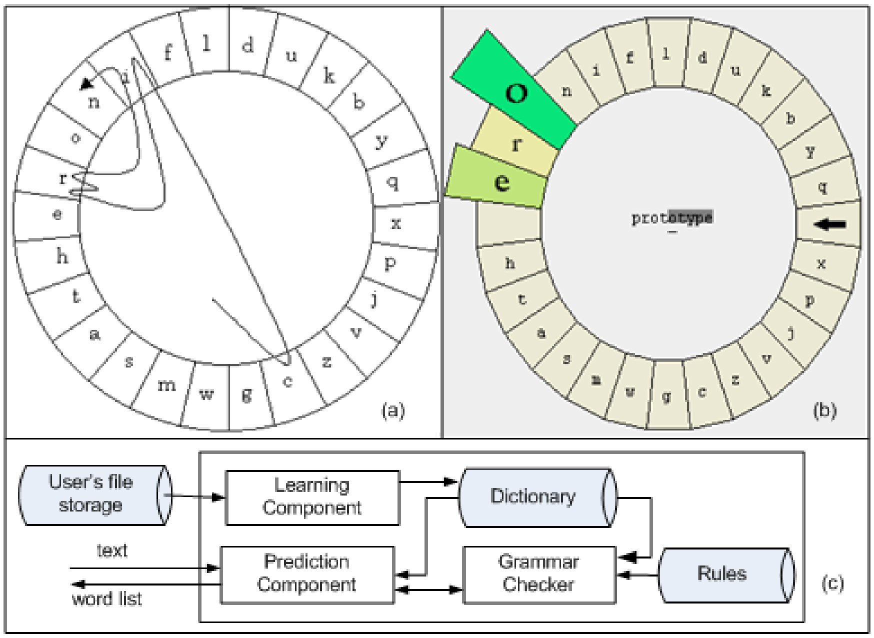

Fig. 2. (a) Classic Cirrin, (b) personalized-adaptive Cirrin, and (c) schematic view of our developed word prediction

flushed to the user's text area. Selecting a backspace on a space will result the previous inputted word back to the middle of the circle. The user can select a word completion with a single tap (in tapping mode) or a left-to-right line motion (in motion mode) in the middle of the circle.

Our developed word prediction consists of several components (Fig. 2(c)). It has two main *dictionaries*, such as (1) a common dictionary(from the BNC) and (2) a user-personal dictionary, which consists of sub-dictionaries for every user's task context (i.e. writing documents, e-mailing, and chatting). They consist of unigram, bigram and trigram list, which include part-of-speech tags and frequency. The *learning component* updates both dictionaries by two ways: (1) extracting inputs during interaction and (2) extracting the user's file storage (scheduled). The *prediction component* generates three lists of suggestions after the first character is inputted, such as (a) from the common dictionary, (b) from the entire personal dictionary, and (c) based on the context of user's task. The probability of a sentence is estimated with the use of Bayes rule, where h_i is the relevant history when predicting a word w_i:

$$P(w_1, w_2, ..., w_n) = \prod_{1}^{n} P(w_i \mid w_1, ..., w_{i-1}) = \prod_{1}^{n} P(w_i \mid h_i) \tag{1}$$

The *grammar checker* excludes syntactically implausible words from the suggestion lists and includes suffix completions, in five steps. First, using Qtag POS tagger [18], it parses the user's input and results the highest probability part-of-speech of each word. Second, this component splits a POS-tagged input into chunks of noun phrase, verb phrase and preposition phrase for detecting noun pluralism and verb tense. Third, it creates all forms for each word in the three suggestion lists from the prediction component. Currently we use thirteen suffixes, such as: "s", "ed", "er", "est", "ly", "able", "full", "less", "ing", "ion", "ive", "ment", and "nest". Using WordNet [6], each new form is verified. Since a word form may be ambiguous and adhere to more forms, all word forms are added to the suggestion lists with the same probability. Four, the grammar checker uses a rule-based approach to check each suggestion whether it is confirmed by grammatical, ungrammatical or out of scope of the grammar. The ungrammatical ones are discarded from the lists. Finally, this component will choose the highest probability word from the context-based dictionary preceded the personal and common dictionary. The personal dictionary will be chosen preceded the common dictionary, if the context-based suggestion list is empty.

6 Conclusion and Discussion

In our study we found that the word completion shows better performance using a relatively small dictionary containing the most frequent words. This may indicate that a personalized task based dictionary can offer a more efficient word completion than a large common dictionary. We believe that this can also imply to the accuracy of the word prediction if syntactically implausible words are also excluded from its prediction space. By this way, besides saving time and energy in inputting, a text entry system can also assist the users in the composition of well-formed text. In addition, the number of user inputs for a desired word can be reduced if the system takes an assumption that a suggested word is rejected after the user selects the next character. Therefore, the user can have a better language coverage since each suggestion word is shown only once.

An adaptive single- and zero-handed Cirrin-based on-screen keyboard with personalized language-based techniques acceleration, which include personalized and adaptive task-based dictionary, frequent character prompting, word completion, and grammar checker with suffix completion, has been developed. It allows both tapping and motion-based input. The system's predictive features enable it to display a syntactically plausible word completion and characters in different sizes and color contrasts according to their relative probabilities.

Acknowledgments. The research reported here is part of the Interactive Collaborative Information Systems (ICIS) project, supported by the Dutch Ministry of Economic Affairs, grant nr: BSIK03024.

References

1. A British National Corpus: Unigrams and Bigrams, Retrieved on January 5, from http://natcorp.ox.ac.uk, (2007).

2. Anson D.K., Moist P., Przywara M., Wells H., Saylor H. and Maxime H.: The Effects of Word Completion and Word Prediction on Typing Rates using On-Screen Keyboards. Proc. of RESNA, Arlington, Virginia: RESNA Press, (2005).
3. Biem A.: Minimum Classification Error Training for Online Handwriting Recognition, IEEE Trans. on Pattern Analysis and Machine Intelligence, IEEE Computer Society, CA, USA, (2006), 7(28):1041–1051.
4. Corrada-Emmanuel A.: (n.d.). Enron E-mail Dataset Research. Retrieved in January 5, 2007, from http://ciir.cs.umass.edu/ corrada/enron/
5. Eriksen B.A. and Eriksen C.W.: Effects of Noise Letters Upon the Identification of a Target Letter in a Non-Search Task. Perception and Psychophysics, (1974), 16:143-149.
6. Fellbaum C.: WordNet - An Electronic Lexical Database. The MIT Press, (1998).
7. Fitrianie S., Datcu D., and Rothkrantz L.J.M.: Human Communication based on Icons in Crisis Environments, HCII, China, LNCS Springer, (2007), to appear.
8. Garay-Vitoria N. and Abascal J.: A Comparison of Prediction Techniques to Enhance the Communication Rate. In C. Stary and C. Stephanidis (eds.), User-Centered Interaction Paradigms for Universal Access in the Information Society, LNCS 3196, Springer-Verlag (2004), 400-417.
9. Heidorn G.: Intelligent Writing Assistant. In R. Dale, H. Moisl, and H. Somers (eds.), A Handbook of Natural Language Processing: Techniques and Applications for the Processing of Language as Text. Marcel Dekker, 2000.
10. MacKenzie I.S., Zhang S.X. and Soukoreff R.W.: Text Entry using Soft Keyboards. Behaviour and Information Technology, (1999), 18:235-244.
11. Magnien L., Bouraoui J.L. and Vigouroux N.: Mobile Text Input with Soft Keyboards - Optimization by Means of Visual Clues, Mobile HCI, Springer-Verlag, (2004), 337-341.
12. Maier A., Haderlein T., and Noth E.: Environmental Adaptation with a Small Data Set of the Target Domain, TSD, LNAI 4188, Berlin, Heidelberg (2006), 431-437.
13. Mankoff J. and Abowd G. D.: Cirrin: A Word-Level Unistroke Keyboard for Pen Input. ACM UIST'98, (1998), 213–214.
14. Isokoski P.: Manual Text Entry - Experiments, Models, and Systems. (Ph.D. thesis), Dept. of Comp. Sciences, University of Tampere, Finland, (2004).
15. Karlson A., Bederson B. and Contreras-Vidal J.: Understanding Single Handed Use of Handheld Devices. In Lumsden Jo (Ed.): Handbook of Research on User Interface Design and Evaluation for Mobile Technology, in press, (2006).
16. Langendorf D.J.: Textware Solution's Fitaly Keyboard V1.0 Easing the Burden of Keyboard Input. WinCELair Review. 1988.
17. Sussman V.: Opening Doors to an Inaccessible World. U.S. News and World Report, September, (1994), 85.
18. Tufis D. and Mason O.: Tagging Romanian Texts - a Case Study for QTAG, a Language Independent Probabilistic Tagger, Proc of LREC. Spain, (1998), 589-596.
19. Ward D.A., Blackwell A. and MacKay D.: Dasher - a Data Entry Interface Using Continuous Gesture and Language Models. ACM UIST, NY: ACM (2000), 129-136.
20. Wood M.E.J and Lewis E.: Windmill The Use of a Parsing Algorithm to Produce Predictions for Disabled Persons. Proc. of Autumn Conference on Speech and Hearing, (1996), 18(9): 315-322.
21. ZetaTalk: Chat Logs - ZetaTalk Live Dec 2001-May 2003, Retrieved in January 5, 2007, from http://www.zetatalk3.com/index/zetalogs.htm.
22. Zhai S. and Kristensson P.-O.: Shorthand Writing on Stylus Keyboard, ACM CHI, (2003), 97-104.