A Reverse-Huffman Algorithm for Text Entry Interface Evaluation

Foad Hamidi
Department of Computer Science and Engineering, York University, 4700 Keele St., Toronto, Ontario, Canada M3J 1P3
fhamidi@cse.yorku.ca

Dr. Melanie Baljko
Department of Computer Science and Engineering, York University, 4700 Keele St., Toronto, Ontario, Canada M3J 1P3
mb@cse.yorku.ca

CHI 2012, May 5–10, 2012, Austin, TX, USA.
Workshop on Designing and Evaluating Text Entry Methods

Abstract
Scanning or soft keyboards display symbols to be selected when highlighted in an order. People with disabilities use these interfaces to compose text by using one or two input actions. We present a *reverse-Huffman algorithm* (RHA) that extracts a *representative latent probability distribution* from a soft keyboard design and evaluates and compares it with other designs using the *Jensen-Shannon Divergence* (JSD).

Keywords
Indirect Text Entry, Reverse-Huffman Algorithm, Jensen-Shannon Divergence, Interface Engineering

ACM Classification Keywords
H.5.2 [Information Interfaces and Presentation]: User Interfaces—Evaluation/methodology.

General Terms
Human Factors, Measurement, Design

Introduction
For many people with motor disabilities, using conventional text entry techniques, such as using the computer keyboard, is difficult due to the high degree of fine motor movement required. *Scanning or soft keyboards* contain a set of *selectables* (W), consisting

of *symbols* (such as letters, digits, the space character, and punctuation marks) and *system commands,* arranged and displayed on a display and highlighted in a specific order. Together with an often customizable input mechanism or *input action*, such as a puff switch or a single button, soft keyboards form an *indirect text composition facility* (TCF). The user can select a highlighted selectable or group of selectables by activating the input action. These systems thus afford an effective method for text composition for people with disabilities and have been in use for many decades [3, 4]. Figure 1 shows a soft keyboard.

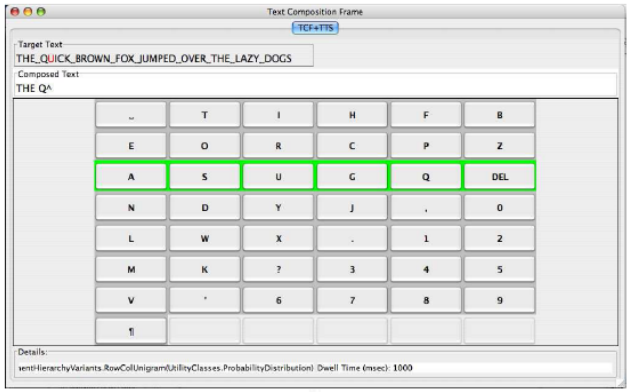

figure 1. Screen shot of a TCF where the selectables are arranged by unigram probabilities and are highlighted in rows and columns.

Using an information theoretic approach, we present a method, the *reverse-Huffman algorithm* (RHA), for the evaluation and design of soft keyboards. Our method is inspired by the *Huffman algorithm* [2] that given a set of symbols, a probability distribution and an out degree value k, produces a coding tree that has the smallest *mean encoding length* (MEL) for each symbol.

A Descriptive Model of TCFs

In previous work [1], we developed a descriptive model of TCFs. In this model, we characterize a soft keyboard using the notion of a *containment hierarchy* (CH). A CH is a directed acyclic graph that expresses the behavior of the system. In the graph, each node is associated with a set of selectables such that each leaf node's set contains a single selectable and each internal node's set contains precisely those selectables that are associated with its children nodes. The interaction is captured by the notion of *focus.* At any given time a single node in the graph is *in focus* corresponding to each highlighting step as seen by the user. In order to select a symbol, the user has to traverse the tree from root to the corresponding leaf. Figure 2 shows a CH corresponding to the soft keyboard described in figure 1.

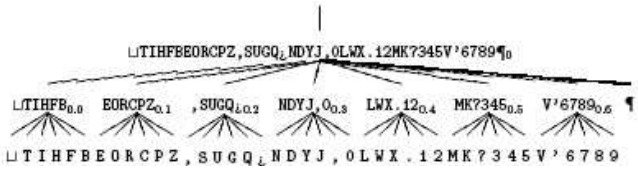

figure 2. A CH corresponding to the TCF shown in figure 1.

For each selectable, the path from the root node of the CH to the corresponding leaf node expresses an *encoding*. The encoding is prefix-free in the sense that each encoding corresponds to an unambiguous selectable. Given a soft keyboard system expressed using the above model the MEL can be calculated by weighing the length of each of these encodings by the relative probability of the selectable in the *empirically*

occurring text. The probability distribution of the *empirically occurring* text, P_W, can be approximated by considering extant text created in similar contexts.

In our work, we show that there is a *representative latent probability distribution,* P'_W, hidden in every CH. Furthermore, this probability distribution expresses the design rational of the CH from an information theoretic point of view and can be used to compare different designs with each other. Extracting P'_W and comparing it with P_W, provides an alternative method for evaluating and comparing soft keyboard designs.

The Reverse-Huffman Algorithm

We have developed a *reverse-Huffman algorithm* (RHA) that extracts a *representative latent probability distribution* (P'_W) from a given CH. RHA takes as its input a CH and an extant *empirically occurring probability distribution,* P_W.

We briefly describe the algorithm: A preliminary step is to transform a given CH into a modified encoding tree in which all internal nodes have an equal number of children by adding *ghost* leafs to them. This modified tree can be viewed as a solution of the Huffman algorithm for some input. Next, a set of linear constraints on the relationships among the probability values associated with the selectables is generated that characterizes the *set of latent probability distributions* that could be input to the Huffman algorithm to result in the modified encoding tree.

The constraints are defined by the following rules. Firstly, the sum of the probabilities of the selectables is equal to 1. Secondly, the probability of each internal node is the sum of the probability of the selectables associated with the leaf nodes that are its children. Next, the selectables are ordered by performing a breadth-first traversal of the tree. This traversal visits leaf-nodes in the order of importance and the corresponding selectables are placed in an ordered set. The constraints express that the probability of each selectable is more than the probability of the selectable following it. These constraints describe a *set of latent probability distributions* that describe the ordering of the selectables in the ordered set.

In the next step, we identify a *representative latent probability distribution,* P'_W, by using the *objective function* of minimizing the distance the absolute difference between the corresponding probability values in P'_W and P_W. The result is further tweaked by minimizing the sum of probability values associated with ghost nodes. The output of RHA is a *representative latent probability distribution,* P'_W, that can now be used to evaluate and compare the given CH with other systems.

Jensen-Shannon Divergence as a Design Metric

The distance between two probability distributions can be calculated using the *Jensen-Shannon Divergence* (JSD). At this point, we can state the central hypothesis of this work: If, as suggested before, the *representative latent probability distribution,* P'_W, expresses the design rational of the CH in terms of what selectables are deemed more important, then the JSD value between P'_W and the *empirically occurring probability distribution,* P_W, can be used to measure and compare the efficiency of the design for text composition.

To test the above hypothesis, we calculated MEL and JSD values for 36 TCF design variants. These designs included 15 variants described by Venkatagiri [5] and 21 variants of the Huffman encoding. We used two different empirically occurring probability distributions: PC_W, derived from a corpus of chat logs and PF_W derived from a corpus of formal English. We calculated the normalized ratio of JSD to MEL values for each of the variants. The normalized ratio of JSD to MEL values is the JSD value over the percentage of improvement of MEL over the worst-case MEL.

We performed a trend line analysis on the results that showed that the two metrics are positively correlated. A trend line coefficient test showed that the correlation is significant for both PC_W and PF_W. Further ANOVA analysis revealed that the correlation between MEL and JSD is not affected by factors such as (1) the derivation technique for the TCF (e.g., Human-based or manually-derived), (2) input probability distribution to the RHA, and (3) the size of the set of selectables.

These results confirm that JSD correlates to MEL and, in combination with RHA, can be used as an alternative metric to MEL. While MEL has been used extensively to analyze text entry methods, it is useful to have JSD as an alternative metric for several reasons. First, JSD is more straightforward to interpret than MEL: it has a lower bound of zero and tends to infinity. Moreover, it is valid to compare the JSD values of very different kinds of TCFs (e.g., in terms of CH structures, keyboard layouts, etc) amongst one another. This affords the investigator to build up a distribution of JSD values, in order to develop an intuition about how to interpret particular JSD values.

Conclusion

We presented a novel method and metric for the evaluation and comparison of soft keyboard designs. The *reverse-Huffman algorithm* (RHA) extracts a *representative latent probability distribution* from a soft keyboard design variant that captures that expresses the design rational in terms of the relative importance of the selectables. The *Jensen-Shannon* Distribution (JSD) is then used to calculate the distance between the latent distribution and an empirically occurring distribution. We have shown that this metric is positively correlated with *mean encoding length* (MEL) and can be used to measure and compare the efficiency of the design variant for text composition.

References

[1] Baljko, M. and Tam, A. Indirect text entry using one or two keys. In *Proc. ASSETS'06,* ACM Press (2006), 18-25.

[2] Huffman, D. A. A method for the construction of minimum-redundancy codes. Proceedings of the Institute for Radio Engineers 40, 9 (1952), 1098-1102.

[3] Lesher, G., Moulton, B., and Higginbotham, D. J. Techniques for augmenting scanning communication. Augmentative and Alternative Communication 14 (1998), 81-101.

[4] MacKenzie, I. S. The one-key challenge: Searching for a fast one-key text entry method. In *Proc. ASSETS'09,* ACM Press (2009), 91-98.

[5] Venkatarigi, H.S. Efficient keyboard Layouts for sequential access in augmentative and alternative communication. Augmentative and Alternative Communication 15, 2 (1998), 126-134.