Keystroke Level model

Olivier Georgeon & Frank Ritter, march 2012
The KLM: HCI’s First Predictive Model


Outline

- Problem statement
- Model
- Empirical validation
- Applications
Problem Statement

• Goal
  – Develop simple model to describe time to do task with a given method on an interactive system

• \( T_{task} = T_{acquire} + T_{execute} \)
  where
  \( T_{task} \) = total time to complete task
  \( T_{acquire} \) = time to select method to complete task
  \( T_{execute} \) = time to perform method

• Model predicts \( T_{execute} \)

• Assume expert users and no errors
Model

- $T_{execute} = \sum$ (time to execute primitive op)
- Primitive operations
  - K  key/button press
  - P  point to target with mouse
  - H  home hands to keyboard or mouse
  - D  draw line with mouse
  - M  mental preparation (pause)
  - R  system response time
Model (cont.)

• Times for primitive operations are predicated from experiments
  – Time to press key/button ranges between
    • 0.08 sec/char    fast typist
    • 0.28 sec/char    average typist
    • 1.2 sec/char     slow typist
## Original parameters

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
<th>Time (s)</th>
</tr>
</thead>
</table>
| K        | PRESS A KEY OR BUTTON  
Pressing a modifier key (e.g., shift) counts as a separate operation. Time varies with typing skill:  
Best typist (135 wpm)  
Good typist (90 wpm)  
Average skilled typist (55 wpm)  
Average non-secretary typist (40 wpm)  
Typing random letters  
Typing complex codes  
Worst typist (unfamiliar with keyboard) | .08  
.12  
.20  
.28  
.50  
.75  
1.20 |
| P        | POINT WITH A MOUSE  
Empirical value based on Fitts' law. Range from .8 to 1.5 seconds. Operator does not include the button click at the end of a pointing operation | 1.10 |
| H        | HOME HAND(S) ON KEYBOARD OR OTHER DEVICE | .40 |
| D\(n_0, l_0\) | DRAW \(n_0\) STRAIGHT-LINE SEGMENTS OF TOTAL LENGTH \(l_0\). Drawing with the mouse constrained to a grid. | \(.9 n_0 + .16 l_0\) |
| M        | MENTALLY PREPARE | 1.35 |
| R(t)     | RESPONSE BY SYSTEM  
Different commands require different response times. Counted only if the user must wait. | \(t\) |
Encoding Method

• Code method as sequence of primitive operations, sum up times
• Example: replace 5 letter word with another
• Editor 1: keyboard based
  – next line MK[(line feed)]
  – substitute command MK[s]
  – enter new word 5K[xxxxx]
  – terminate arg MK[(return)]
  – enter old word 5K[zzzzz]
  – terminate arg MK[(return)]
  – terminate command K[(return)]

• $T_{execute} = 4t_M + 15t_K = 8.4 \text{ sec}$
Placement of M’s

rule 0: place M before all K’s not part of arg strings and before all P’s that select commands
for each M do

rule 1: if operator fully anticipated, delete M
(e.g. PMK → PK)
rule 2: if string of MK’s belongs to cognitive unit, delete all M’s but first
rule 3: if K is redundant terminator, delete M
rule 4: if K terminates constant string (i.e. command), delete it
if K terminates variable string (i.e. arg), keep it
Placement of M’s

Example: Mouse-based Editor 2 sequence

\[
\]

⇒

\[
H[m]P[word]MK[y]H[k]MK[R]5K[\ldots]MK[(esc)]
\]

rule 0:

\[
\uparrow \quad \uparrow \quad \uparrow
\]

M

M

M

⇒

\[
H[m]P[word]\text{delete}H[k]MK[R]5K[\ldots]MK[(esc)]
\]

rule 1:

delete

⇒

\[
H[m]P[word]K[y]H[k]MK[R]5K[\ldots]MK[(esc)]
\]
Placement of M’s

Example: Keyboard-based Editor 1 sequence

\[
\begin{align*}
\uparrow \uparrow \uparrow \uparrow \uparrow \\
M & M & M & M & M \\
\end{align*}
\]

rule 0:

\[
\begin{align*}
\uparrow & \text{delete}
\end{align*}
\]

rule 3:
Empirical Validation

• Conducted experiment on real systems to compare predictions to observations
• 14 tasks on 3 types of systems
  – text editors
  – graphics systems
  – command executers
• Users
  – practice using system
  – were tested to determine typing and drawing speed
Observation vs. prediction
Some Recent Applications of the KLM


The KLM and Text Entry

• The KLM and Text Entry
  – The original experiment did not include a task such as:
    – Enter a 43-character phrase of text

• Why?
  – It’s too simple for the KLM
  – Prediction reduces to $43 \times tK$
  – $tK$ determined for each user in a pre-test
  – The KLM task just confirms the pre-test
T9 Entry of “beep”

2 a
23 be
233 bed
2337 beer
2337n beds
2337nn adds
2337nnn bees
2337nnnn beep
2337nnnn0 beep_
Model for T9 Entry of “beep”

$2 3 3 7 M_P n M_P n M_P n M_P n M_P n M_P 0$

$M_P$ - the time for performing a physical match between a stimulus (the presented word) and a code stored in the user’s short-term memory (the desired word).
Summary

• $T_{task} = T_{acquire} + T_{execute}$

• $T_{execute} = \text{Key} + \text{Point} + \text{Home} + \text{Draw} + \text{Mental} + \text{System}$

• Very low level

• Simplistic

• Famous reference

• Still useful 25 years later