A Computational Model of Routine Procedural Memory

Frank Tamborello
Department of Psychology
Rice University
Houston, TX 77005
tambo@rice.edu
http://chil.rice.edu/
Overview

- Very Brief Introduction
- Two Experiments, Very Briefly
- ACT-R Model
Context
This feature is important for a number of reasons, one being that deliberate conscious control is often too slow to allow for the speed of many of our skilled behaviors. Additionally, trigger conditions can sometimes overwhelm influences from the SAS, as in the Stroop task. Perhaps most importantly, Norman and Shallice argue, is that certain types of errors seem to indicate that deliberate conscious control is not always required for action. In particular, "capture errors" occur when a person begins one task, and through inattention and/or distraction switches, before completion of the original task, to a new task that is at least as familiar as the original task. Reason and Mycielska (1982) documented an example of a capture error in a diary study they conducted.

Figure 2. The overall system: vertical and horizontal threads. When attention to particular tasks is required, vertical thread activation comes into play. Attention operates upon schemas only through manipulation of activation values, increasing the values for desired schemas, decreasing (inhibiting) the values for undesired ones. Motivational variables are assumed to play a similar role in the control of activation, but working over longer time periods. To emphasize that several tasks are usually active, with the individual components of each task either being simultaneous or overlapping in time, this figure shows five different horizontal threads. Some means of selecting the individual schemas at appropriate times while providing some form of conflict resolution becomes necessary. The interactions among the
Each step of processing carries information about the state of the system at the previous time step, thus the system is driven by a number of actions. Open arrows indicate that every unit in the sending layer is connected to every unit in the receiving layer.

**Simple Recurrent Network (SRN)**

The SRN architecture consists of several components:

- **Actions**
- **Internal representation**
- **Perceptual input**

The network receives input from the environment, processes it, and outputs actions. The internal representation is updated based on the input and actions, and this process iterates over time. The network is designed to model sequential behavior and decision-making processes.

Botvinick and Plaut echo Lashley’s hypothesis that some actions appear in multiple scenes, and they propose that this can be handled by the SRN’s ability to maintain and update internal representations over time. The SRN is a computational model that includes a variety of tasks and is designed to focus on the physical world and the actions of objects within it.
GOMS

- **GOAL: EDIT-MANUSCRIPT**
  - **GOAL: EDIT-SUBTASK**  *repeat until no more subtasks*
    - **GOAL: ACQUIRE-SUBTASK**
      - GET-NEXT-PAGE  *if at end of manuscript page*
      - GET-NEXT-TASK
    - **GOAL: EXECUTE-SUBTASK**
      - **GOAL: LOCATE-LINE**
        - [select: USE-QUOTED-STRING-METHOD
          - USE-LINEFEED-METHOD]
      - **GOAL: MODIFY-TEXT**
        - [select: USE-SUBSTITUTE-COMMAND
          - USE-MODIFY-COMMAND
          - VERIFY-EDIT]
Inputs:
- Knowledge
  - IF-THEN rules (termed “productions”)
  - Declarative knowledge (“chunks”)
  - Subsymbolic parameters
- Simulated task environment/world

Output: Time-stamped behavior sequence
Experiment Overview

- Task is a routine procedure
- Subjects trained approximately one week before
- Concurrent working memory task given
Experiment 1 Condition

- static, intervening subtask
- procedure change, pre-change
- procedure change, post-change
- non-intervention
- semantic control

Mean Total Error Rate

0.00
0.02
0.04
0.06
0.08
0.10
0.12
0.14
0.16

Mean Total Error Rate
Experiment 2 Condition

- Static, different-scanner
- Static, same-scanner
- Change procedure, pre-change
- Change procedure, post-change
- Static subtask reordering

Mean Total Error Rate

<table>
<thead>
<tr>
<th>Condition</th>
<th>Mean Total Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static, different-scanner</td>
<td>0.12</td>
</tr>
<tr>
<td>Static, same-scanner</td>
<td>0.04</td>
</tr>
<tr>
<td>Change procedure, pre-change</td>
<td>0.08</td>
</tr>
<tr>
<td>Change procedure, post-change</td>
<td>0.14</td>
</tr>
<tr>
<td>Static subtask reordering</td>
<td>0.06</td>
</tr>
</tbody>
</table>
The Model

- Model Goal: Simulate error rates across conditions and trial types
  - 4 conditions
  - 14 trial types total
  - not just error generation, but also recovery
- Highest human SEM error rate = 0.0415
  - model should do no worse across the board
Basic Model Functioning

Retrieve

Find

Move

Act

Verify

Specify next action

Error Recovery

Try again to retrieve the action

Retrieve another action

Error?

YES

NO

Try again to retrieve the action

Specify next action

Error Recovery

Retrieve another action
Procedure Change

1. **Current step = flagged step?**
   - YES: Retrieve New Procedure’s Step
   - NO: Retrieve

2. **Retrieve**
   - Find
   - Move
   - Act

3. **Verify**
   - Specify next action

4. **Error Recovery**
   - Try again to retrieve the action
   - Retrieve another action

5. **Error?**
   - YES
   - NO: Repeat from step 2
![Bar chart showing mean error rate for different conditions.](image)

- **Jammer, Experiment 2 Condition**

- **Conditions**: static, different-scanner, static, same-scanner, change procedure, pre-change, change procedure, post-change, static subtask reordering

- **Legend**:
  - **humans**
  - **model**

- **Mean Error Rate**
  - Values range from 0 to 0.16.
Model Discussion

- Discrete, hierarchical goals
  - governed basic behavior
  - enabled extensible behavior
Basic Model Functioning

Error Recovery
- Try again to retrieve the action
- Retrieve another action

Verify
- Specify next action

Act
- Error?
  - YES
  - NO

Retrieve
- Find
- Move

Retrieve another action
Try again to retrieve the action
Procedure Change

Current step = flagged step?

NO

Retrieve

Find

Move

Act

YES

Retrieve New Procedure's Step

Verify

Specify next action

Error Recovery

Try again to retrieve the action

Retrieve another action

Error?

NO

YES
Model Discussion

- No quantitative, multi-condition error models in literature
- Same model mechanisms across
  - 4 between-subjects conditions
  - 14 trial types
Future Work

- Extend model
  - Step-level error
  - Step completion time
- Model training, too
General Discussion

- Hierarchical, discrete goal representation matters
  - …for changing circumstances
  - …for error recovery
  - …like CSM

- Botvinick & Plaut’s connectionist model too narrow
  - No postcompletion errors
  - No error recovery
  - No adaptation of old procedures to new circumstances
Acknowledgments

- Mike
- Carissa Chang & Adam Purtee
- Rick Cooper & Jay McClelland
- Kristen
Thank you!

- Questions?