

Functional Interactions between Memory and Recognition Judgments

Justin Li

Computer Science and Engineering
University of Michigan
justinnh@umich.edu

2012-06-20

Memories are Big and Slow

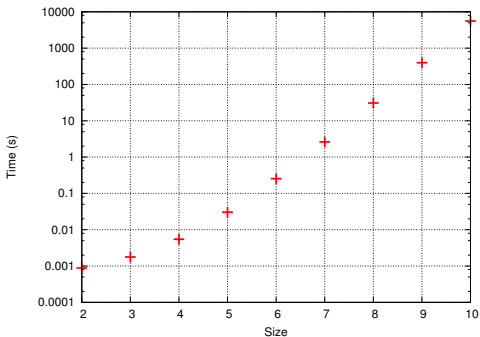
- ▶ Long-lived agents accumulate large amounts of knowledge

Memories are Big and Slow

- ▶ Long-lived agents accumulate large amounts of knowledge
- ▶ Memory searches are slow and potentially unbounded

Memories are Big and Slow

- ▶ Long-lived agents accumulate large amounts of knowledge
- ▶ Memory searches are slow and potentially unbounded



Memories are Big and Slow

- ▶ Long-lived agents accumulate large amounts of knowledge
- ▶ Memory searches are slow and potentially unbounded
- ▶ Comprehensive searches in every scenario is infeasible

Memories are Big and Slow

- ▶ Long-lived agents accumulate large amounts of knowledge
- ▶ Memory searches are slow and potentially unbounded
- ▶ Comprehensive searches in every scenario is infeasible

Problem: How can the agent efficiently retrieve knowledge from an ever-growing memory?

Memories are Big and Slow

- ▶ Long-lived agents accumulate large amounts of knowledge
- ▶ Memory searches are slow and potentially unbounded
- ▶ Comprehensive searches in every scenario is infeasible

Problem: How can the agent efficiently retrieve knowledge from an ever-growing memory?

One solution: Only search memory if there is high probability of a successful retrieval

Memories are Big and Slow

- ▶ Long-lived agents accumulate large amounts of knowledge
- ▶ Memory searches are slow and potentially unbounded
- ▶ Comprehensive searches in every scenario is infeasible

Problem: How can the agent efficiently retrieve knowledge from an ever-growing memory?

One solution: Only search memory if there is high probability of a successful retrieval

In particular: Use the *recognition judgment* as a cheap proxy for whether a retrieval will be successful

Outline

- ▶ What is recognition?
- ▶ How do we calculate recognition judgments in Soar?
- ▶ How do we represent recognition judgments in Soar?
- ▶ How do we evaluate this system?
- ▶ How does the system performance in a WSD task?

Recognition in Long-Term Memories

Recognition A binary signal indicating a previously (un)seen feature (WME), with respect to specific memories

Recognition in Long-Term Memories

Recognition A binary signal indicating a previously (un)seen feature (WME), with respect to specific memories

Continue with memory retrieval *only if the cue is recognized.*

Recognition in Long-Term Memories

Recognition A binary signal indicating a previously (un)seen feature (WME), with respect to specific memories

Continue with memory retrieval *only if the cue is recognized.*

Key idea: reuse and build off long-term memory data structures and processes

Recognition in Long-Term Memories

Recognition A binary signal indicating a previously (un)seen feature (WME), with respect to specific memories

Continue with memory retrieval *only if the cue is recognized.*

Key idea: reuse and build off long-term memory data structures and processes

- ▶ EpMem reuses WME ID, necessary for storage optimizations

Recognition in Long-Term Memories

Recognition A binary signal indicating a previously (un)seen feature (WME), with respect to specific memories

Continue with memory retrieval *only if the cue is recognized.*

Key idea: reuse and build off long-term memory data structures and processes

- ▶ EpMem reuses WME ID, necessary for storage optimizations
- ▶ SMem reuses attribute counters, necessary for retrieval optimizations

Recognition in Long-Term Memories

Recognition A binary signal indicating a previously (un)seen feature (WME), with respect to specific memories

Continue with memory retrieval *only if the cue is recognized.*

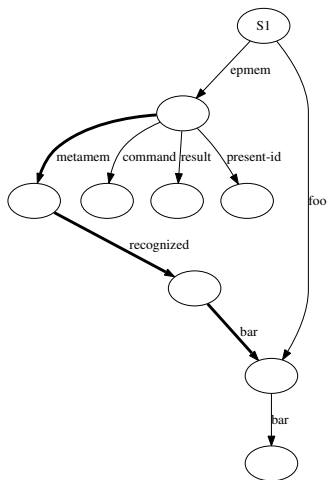
Key idea: reuse and build off long-term memory data structures and processes

- ▶ EpMem reuses WME ID, necessary for storage optimizations
- ▶ SMem reuses attribute counters, necessary for retrieval optimizations

Both judgments are calculated automatically at EpMem storage time

How to Represent Recognition Information?

- Representation 1: WME from $\hat{\{s, ep\}}_{mem}$ to parent, with identical attribute names



How to Represent Recognition Information?

- ▶ Representation 1: WME from $\hat{\{s, ep\}}$ mem to parent, with identical attribute names
- ▶ Representation 2: Extend RETE representation to include bit-array of recognition information

```

sp {recognition*rule
    (state <s> ^foo.bar <b>
        :epmem-recognized)
    ...
-->
    ...
}

```

Evaluation Criteria

Evaluation Criteria

- ▶ **Frugality** By how much does recognition slow down the decision cycle on average?

Evaluation Criteria

- ▶ **Frugality** By how much does recognition slow down the decision cycle on average?
- ▶ **Predictiveness** How well does recognition reflect of whether knowledge exists in memory?

Evaluation Criteria

- ▶ **Frugality** By how much does recognition slow down the decision cycle on average?
- ▶ **Predictiveness** How well does recognition reflect of whether knowledge exists in memory?
- ▶ **Interoperability**

Evaluation Criteria

- ▶ **Frugality** By how much does recognition slow down the decision cycle on average?
- ▶ **Predictiveness** How well does recognition reflect of whether knowledge exists in memory?
- ▶ **Interoperability**
 - ▶ **Task Performance** Is task performance negatively affected?

Evaluation Criteria

- ▶ **Frugality** By how much does recognition slow down the decision cycle on average?
- ▶ **Predictiveness** How well does recognition reflect of whether knowledge exists in memory?
- ▶ **Interoperability**
 - ▶ **Task Performance** Is task performance negatively affected?
 - ▶ **Memory Retrievals** Is the number of retrievals reduced by using recognition?

Evaluation Criteria

- ▶ **Frugality** By how much does recognition slow down the decision cycle on average?
- ▶ **Predictiveness** How well does recognition reflect of whether knowledge exists in memory?
- ▶ **Interoperability**
 - ▶ **Task Performance** Is task performance negatively affected?
 - ▶ **Memory Retrievals** Is the number of retrievals reduced by using recognition?
 - ▶ **Running Time** How is overall running time affected?

The Word Sense Disambiguation (WSD) Task

- ▶ Problem: Assign meaning (*sense*) to polysemous word, given sentence context:
 - ▶ He *ran* the race
 - ▶ He *ran* the code

- ▶ Given: sentence parse tree and ambiguous word
- ▶ Provide: predicted sense of the word
- ▶ Receive: yes/no; correct sense of the word

- ▶ Run on all sentences in SemCor dataset

WSD Agent Design

Agent begins with empty episodic and semantic memories

1. Query episodic memory using parse context
 - ▶ If retrieval succeeds, use retrieved meaning
2. Else query semantic memory (without context)
 - ▶ If retrieval succeeds, use retrieved meaning
3. Else return “Don’t Know”

On feedback, store true meaning in memories

When using recognition, don’t query memory if recognition fails

Is Recognition Frugal?

Agent always returns “Don’t Know”

- ▶ **Baseline:** No recognition judgment is made
- ▶ **Calculated:** Recognition judgment is calculated but *not* represented to the agent
- ▶ **Represented:** Recognition judgment is calculated *and* represented to the agent

Condition	WME (msec)	RETE (msec)
Baseline		
Calculated		
Represented		

Is Recognition Frugal?

Agent always returns “Don’t Know”

- ▶ **Baseline:** No recognition judgment is made
- ▶ **Calculated:** Recognition judgment is calculated but *not* represented to the agent
- ▶ **Represented:** Recognition judgment is calculated *and* represented to the agent

Condition	WME (msec)	RETE (msec)
Baseline	0.137	
Calculated	0.142	
Represented	0.654	

Is Recognition Frugal?

Agent always returns “Don’t Know”

- ▶ **Baseline:** No recognition judgment is made
- ▶ **Calculated:** Recognition judgment is calculated but *not* represented to the agent
- ▶ **Represented:** Recognition judgment is calculated *and* represented to the agent

Condition	WME (msec)	RETE (msec)
Baseline	0.137	0.184
Calculated	0.142	0.191
Represented	0.654	0.196

Is Recognition Frugal?

Agent always returns “Don’t Know”

- ▶ **Baseline:** No recognition judgment is made
- ▶ **Calculated:** Recognition judgment is calculated but *not* represented to the agent
- ▶ **Represented:** Recognition judgment is calculated *and* represented to the agent

Condition	WME (msec)	RETE (msec)
Baseline	0.137	0.184
Calculated	0.142	0.191
Represented	0.654	0.196

These results are highly domain specific

Is Recognition Predictive?

- ▶ 100% predictive in WSD task
 - ▶ Neither false positives nor false negatives

Is Recognition Predictive?

- ▶ 100% predictive in WSD task
 - ▶ Neither false positives nor false negatives
- ▶ False positives are possible in general

How Does Recognition Affect Task Performance?

Is task performance negatively affected?

How Does Recognition Affect Task Performance?

Is task performance negatively affected?

- ▶ Task performance remains the same

How Does Recognition Affect Task Performance?

Is task performance negatively affected?

- ▶ Task performance remains the same

Does recognition reduce number of retrievals?

How Does Recognition Affect Task Performance?

Is task performance negatively affected?

- ▶ Task performance remains the same

Does recognition reduce number of retrievals?

- ▶ Episodic memory retrievals decrease by 70%
- ▶ Semantic memory retrievals decrease by 40%

How Does Recognition Affect Task Performance?

Is task performance negatively affected?

- ▶ Task performance remains the same

Does recognition reduce number of retrievals?

- ▶ Episodic memory retrievals decrease by 70%
- ▶ Semantic memory retrievals decrease by 40%

How is overall running time affected?

How Does Recognition Affect Task Performance?

Is task performance negatively affected?

- ▶ Task performance remains the same

Does recognition reduce number of retrievals?

- ▶ Episodic memory retrievals decrease by 70%
- ▶ Semantic memory retrievals decrease by 40%

How is overall running time affected?

- ▶ Agent completes WSD task with 4x less time

How Does Recognition Affect Task Performance?

Is task performance negatively affected?

- ▶ Task performance remains the same

Does recognition reduce number of retrievals?

- ▶ Episodic memory retrievals decrease by 70%
- ▶ Semantic memory retrievals decrease by 40%

How is overall running time affected?

- ▶ Agent completes WSD task with 4x less time

These results are highly domain specific

Nuggets and Coal

Nuggets

- ▶ Implemented recognition judgments for semantic and episodic memory
- ▶ Expanded RETE and Soar syntax to match binary bits
- ▶ Evaluation on WSD task
- ▶ Possibilities for other meta-memory information

Coal

- ▶ Only evaluated on a single task
- ▶ Don't (yet) entirely understand effect of RETE changes
- ▶ Slightly *hacky* implementation; not ready for integration into trunk

Thank You

For details: Li, Derbinsky, Laird (2012). Functional Interactions Between Memory and Recognition Judgments. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, Toronto, Canada.

Recognition in Long-Term Memories

	Not Recognized by EpMem	Recognized by EpMem
Not Recognized by SMem	new feature	old, un-stored feature
Recognized by SMem	old feature in new context	old feature in old context

Recognition in Long-Term Memories

Key idea: reuse and build off long-term memory data structures and processes

Recognition in Long-Term Memories

Key idea: reuse and build off long-term memory data structures and processes

WM → LTM
Translation



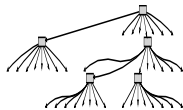
Recognition in Long-Term Memories

Key idea: reuse and build off long-term memory data structures and processes

WM \rightarrow LTM
Translation



Database
Indexing



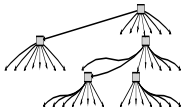
Recognition in Long-Term Memories

Key idea: reuse and build off long-term memory data structures and processes

WM → LTM
Translation



Database
Indexing



SQL
Database



Recognition in Long-Term Memories

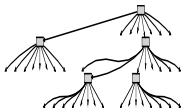
Key idea: reuse and build off long-term memory data structures and processes

WM → LTM
Translation



EpMem reuses WME ID,
necessary for storage
optimizations

Database
Indexing



SQL
Database



Recognition in Long-Term Memories

Key idea: reuse and build off long-term memory data structures and processes

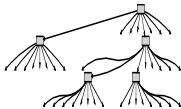
WM → LTM
Translation



EpMem reuses WME ID,
necessary for storage
optimizations



Database
Indexing



SMem reuses attribute
counters, necessary for
retrieval optimizations



SQL
Database



Recognition in Long-Term Memories

Key idea: reuse and build off long-term memory data structures and processes

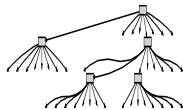
WM → LTM
Translation



EpMem reuses WME ID,
necessary for storage
optimizations



Database
Indexing



SMem reuses attribute
counters, necessary for
retrieval optimizations



SQL
Database

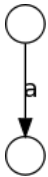


Both judgments are
calculated automatically at
EpMem storage time

Errors in Recognition

SMem

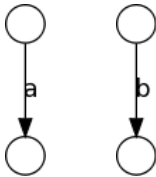
EpMem



Errors in Recognition

SMem

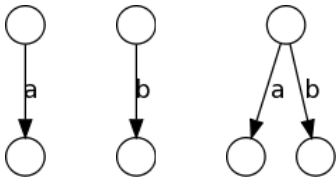
EpMem



Errors in Recognition

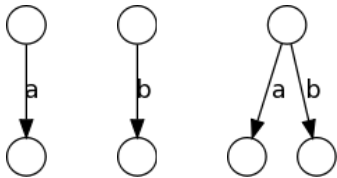
SMem

EpMem



Errors in Recognition

SMem

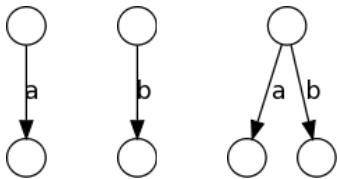


EpMem

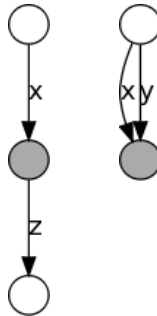


Errors in Recognition

SMem

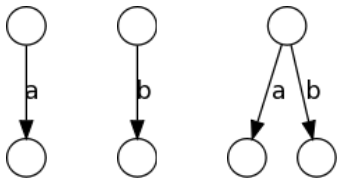


EpMem



Errors in Recognition

SMem



EpMem

