Functional Integration of Concept Semantic Memory and Probabilistic Category Learning in Soar

Yongjia Wang

University of Michigan

Outline

- Definitions & Motivation
- Design and Implementation of Concept Semantic Memory
- Evaluations with Functional Integration Models
- Nuggets and Coal

What is semantic knowledge

- General knowledge independent of specific context
  - Contrast to episodic memory, which is tied to specific context/experience
  - Support sub-symbolic learning: learn consistent prototypes from variations of specific instances
- Semantic knowledge includes many kinds of general knowledge
  - Concepts about concrete things
    - Food, tools, materials …
  - Concepts about abstract things
    - Relation, emotion …
  - Events, facts and information
  - …
- The type of semantic knowledge in this research
  - Functional category knowledge of concrete objects
  - Stored in Concept Semantic Memory

Functional Category Knowledge of Concrete Objects

- Functional category knowledge is one specific form of fundamental semantic knowledge
  - Based on simplest scenario where a single agent interacts with the environment
  - Functional properties: related to direct physical interactions of an embodied agent
  - Objects within the same functional category shares more functional properties than objects belong to different functional categories
- Generalization
  - *although interactions with the world takes place at the level of individual objects, much reasoning takes part at the level of categories* (Russell & Norvig, AIMA)
- Learning approach
  - Knowledge engineering is implausible for more challenging domains

Motivational Category Learning Example – Hunting









Hunt or not?
How to hunt?

Motivational Category Learning Example – Hunting

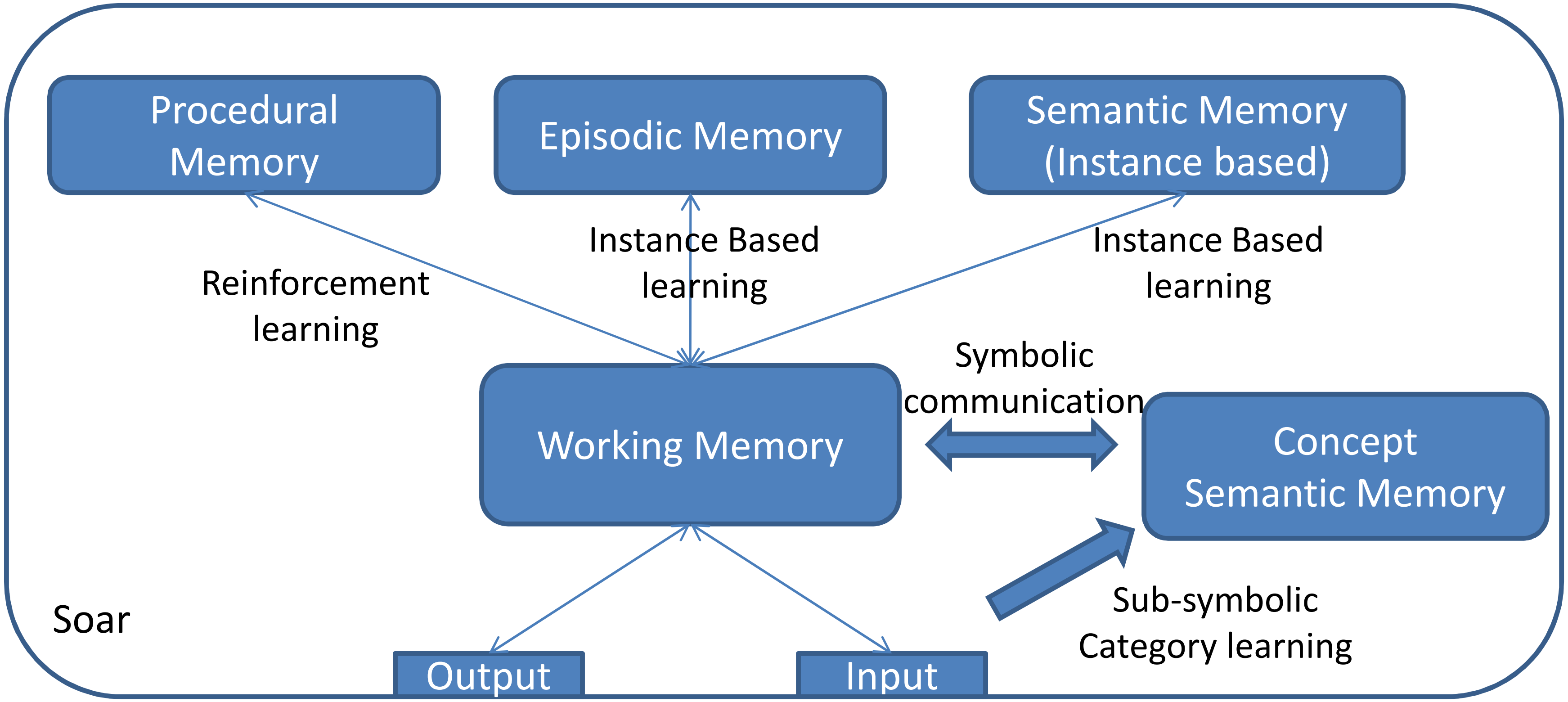
Outline

- Definitions & Motivation
- **Design and Implementation of Concept Semantic Memory**
- Evaluations with Functional Integration Models
- Nuggets and Coal

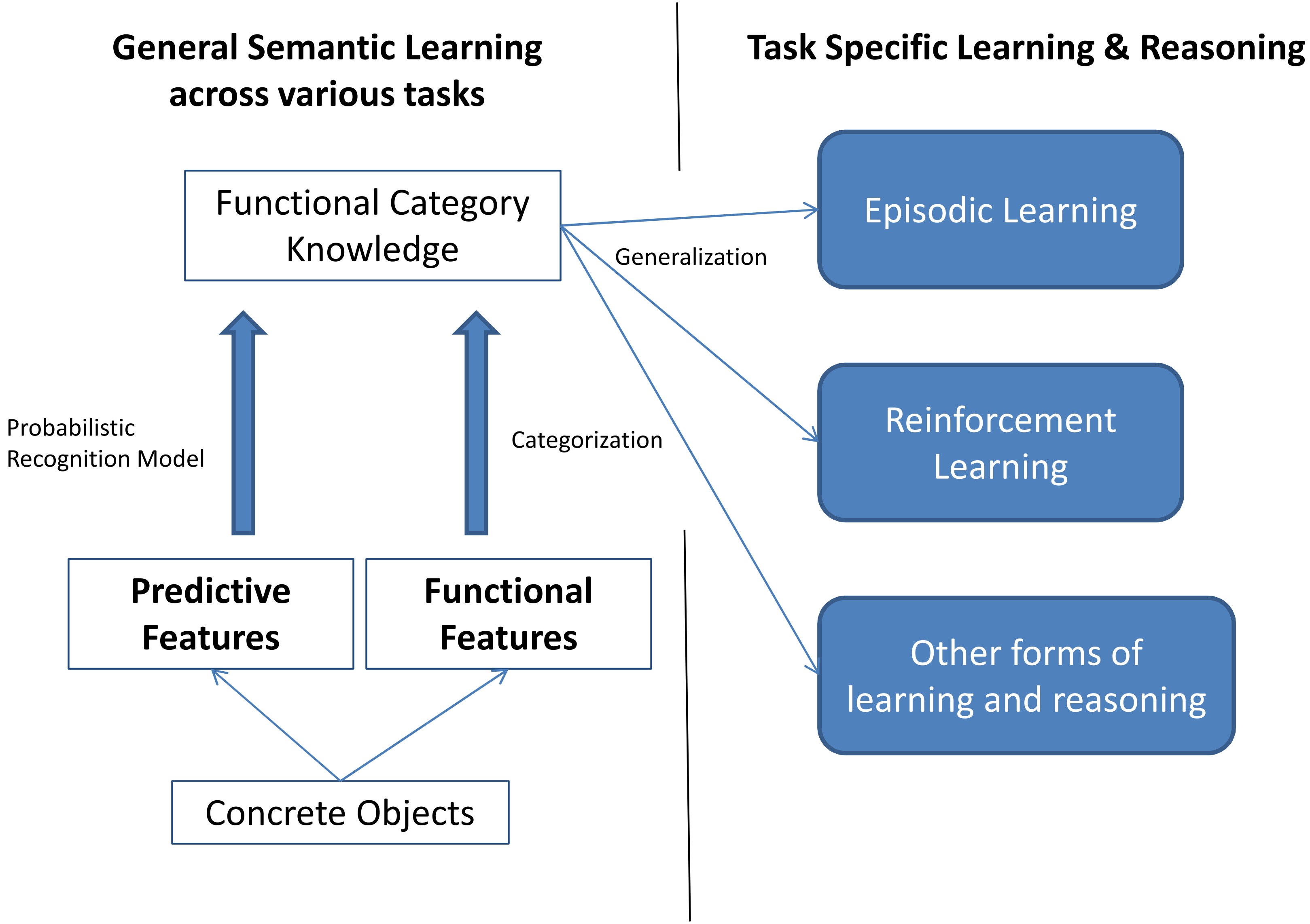
Requirements of Semantic Category Learning

- Incremental learning, noise tolerance (sub-symbolic)
- Integration with diverse knowledge sources
  - Episodic learning
  - Reinforcement learning
- Functionally meaningful (semantics)
  - Categorization must be based on functional properties (related to direct physical interactions)
  - Same object may be used for different purposes, therefore need multiple ways of categorization
    - Example:
      - Categorize swordfish as food
      - Categorize swordfish as weapon

Architectural Design



Functional Integration



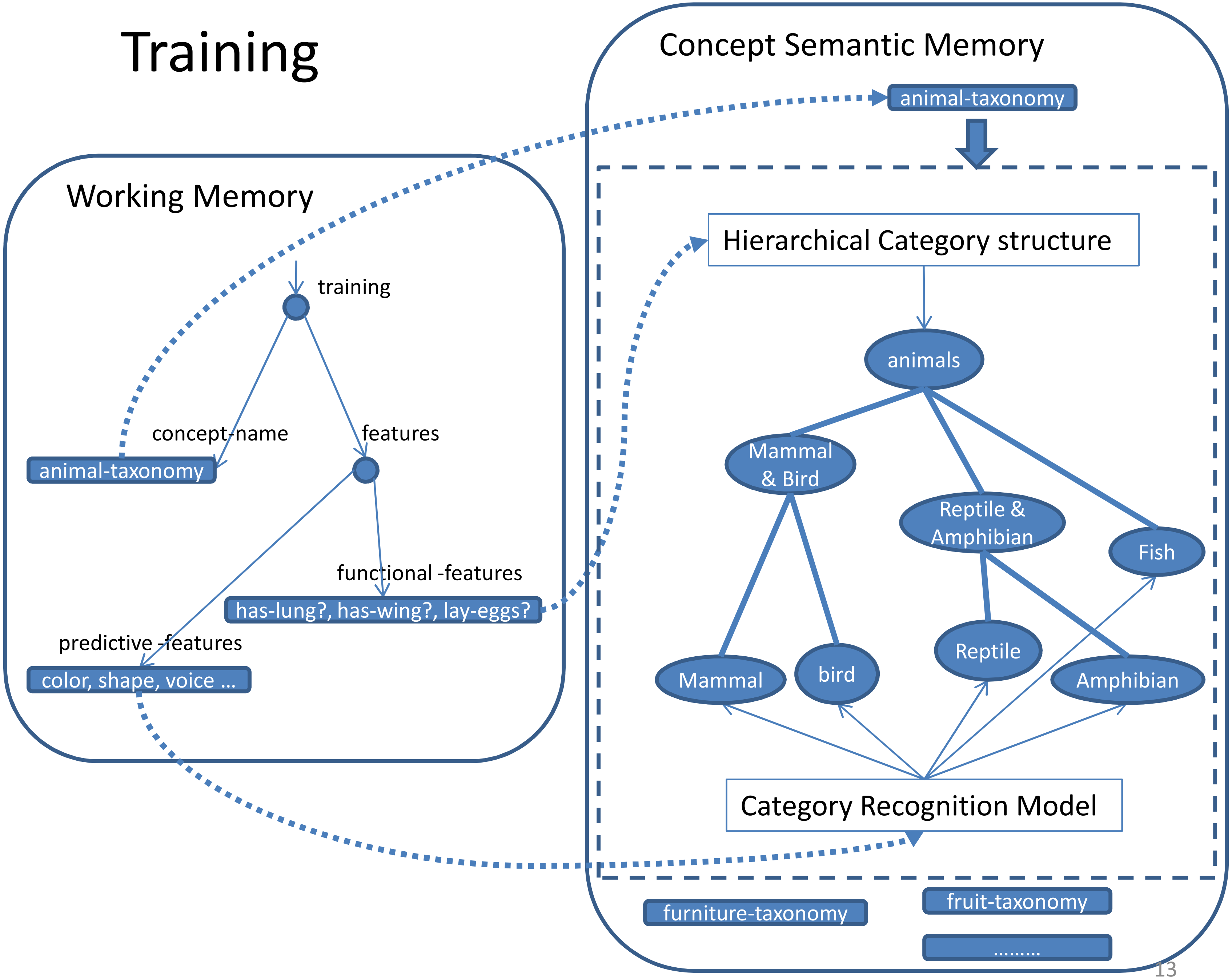
Functional Features and Predictive Features

Functional Features	Predictive Features
Properties directly related to actions	Indirectly related to functional properties
Example: shape and surface texture of an object to be gripped by a robotic arm agent	Example: Color of the object to be gripped
More “expensive” to observe: require interaction	“Cheaper” to observe: ranged sensors
Prior knowledge: define categorization criteria	Learning: select relevant features and ignore irrelevant features in order to predict functional features more accurately

Interface of Concept Semantic Memory to the Architecture

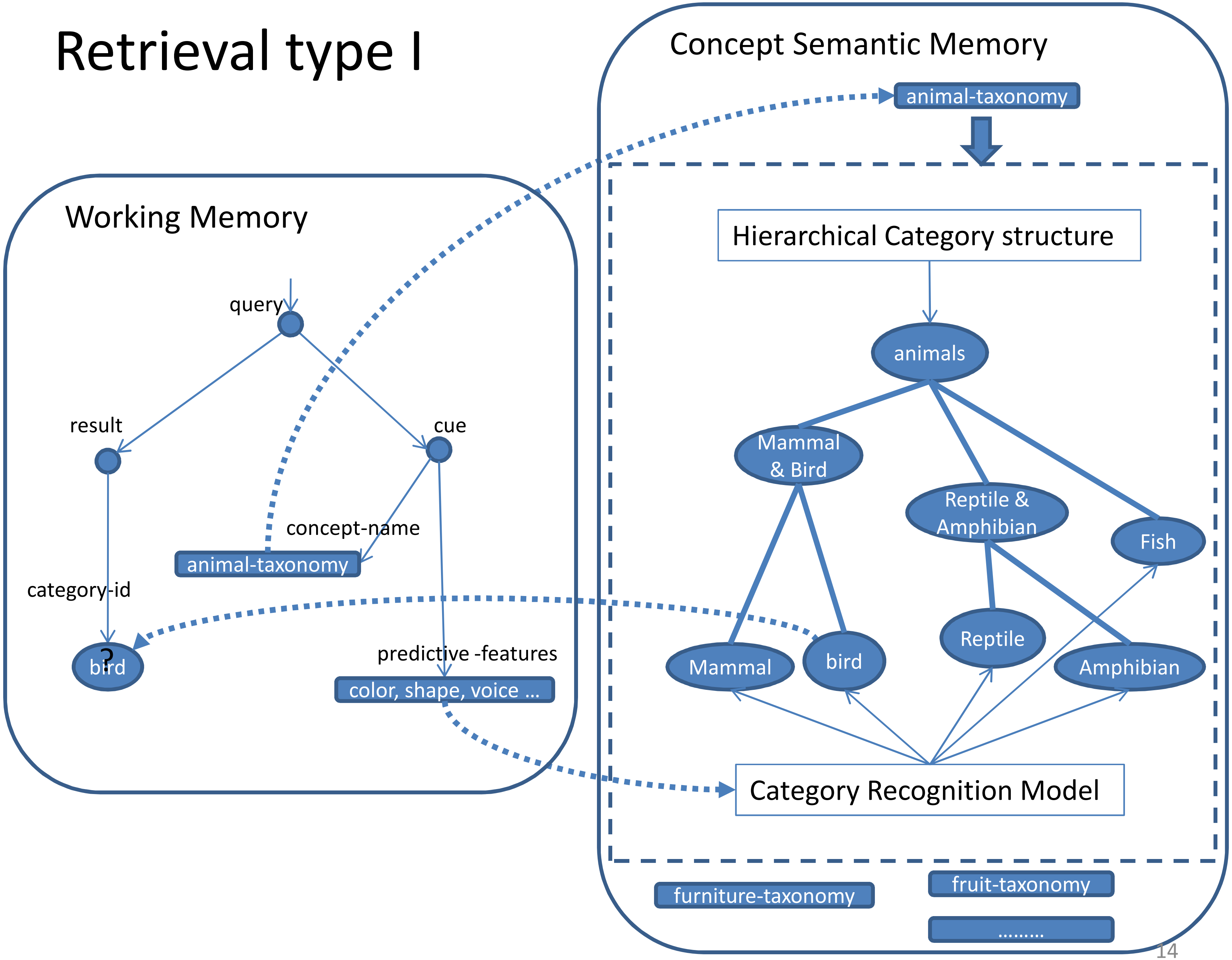
- Train the system with an object for a specific categorization criteria
  - Example:
    - Categorize swordfish (object) as food (criteria)
    - Categorize swordfish (object) as weapon (criteria)
- Retrieve the symbolic category given predictive features
- Retrieve the functional features given predictive features

Training

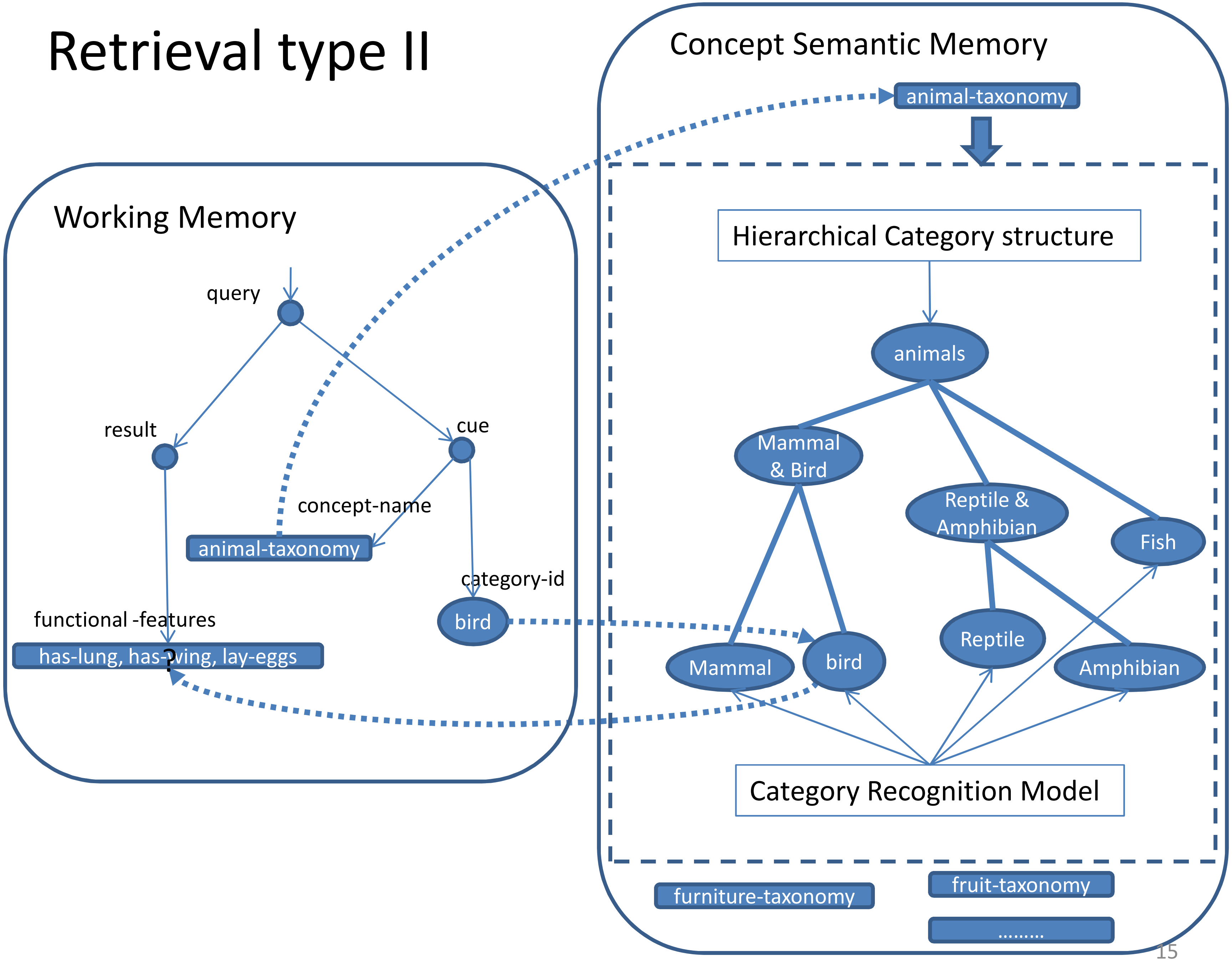


Working Memory
training
concept-name
features
animal-taxonomy
functional -features
has-lung?, has-wing?, lay-eggs?
predictive -features
color, shape, voice ...
Concept Semantic Memory
animal-taxonomy
Hierarchical Category structure
animals
Mammal & Bird
Reptile & Amphibian
Fish
Mammal
bird
Reptile
Amphibian
Category Recognition Model
furniture-taxonomy
fruit-taxonomy
....

Retrieval type I

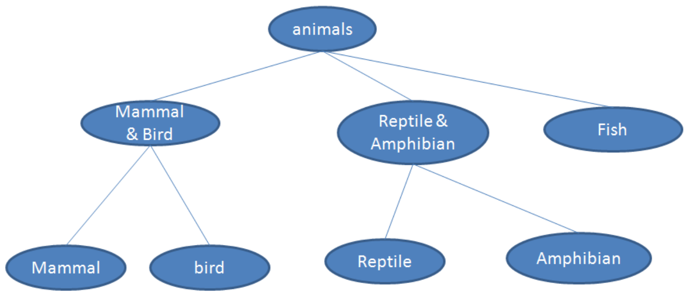


Retrieval type II



Hierarchical Clustering Algorithm

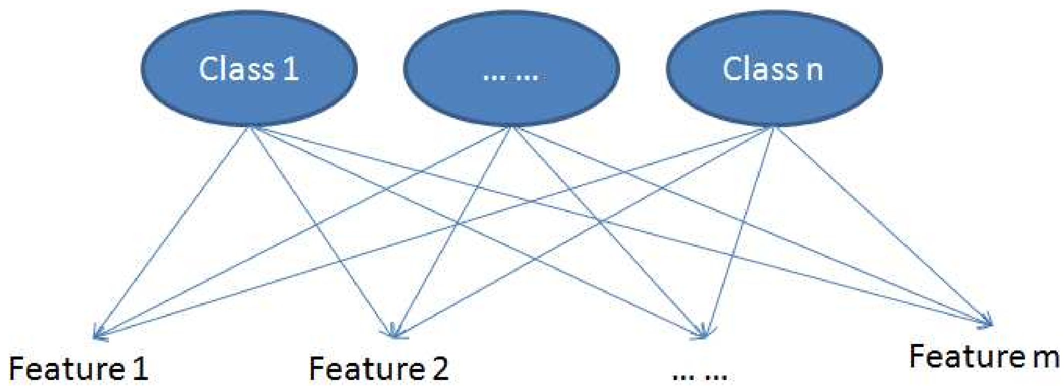
- Adapted from COBWEB (D. Fisher, 1987)
- Incremental learning
- Create hierarchical category structure
- Can deal with several representation forms
  - Nominal features
  - Numeric features
  - Relational structural features



- Robust (noise tolerant)

Category Recognition Model

- Right now we use Naïve Bayesian Classifier
  - Incremental learning
  - A set of “basic level categories” (classes)
  - Complete independency among features give class label
  - Handles numeric features



- Can use more sophisticated models

Outline

- Definitions & Motivation
- Design and Implementation of Concept Semantic Memory
- **Evaluations with Functional Integration Models**
- Nuggets and Coal

Evaluations with Functional Integration Models

- Evaluation Task
  - Hunting task
- Functional Integration Models
  - Integration with Episodic memory
    - Not tested with real Soar Episodic Memory yet
    - Real integration will involve subtle technical issues
  - Integration with Reinforcement learning
    - Tested with real Soar-RL

The Hunting Task

- There are two functional types of objects in the world, both have diverse sub-types (requires category learning)
  - Animals
  - Weapons
- There are complex interactions between the two functional types of objects
  - Certain animal is only huntable with certain weapons
  - Goal is to learn to make the correct decision based on experience

The Hunting Domain Data

Weapon features (both functional and predictive)

Name	type	weight	power-damage	max-range
Sling	ranged	1~2	1~1.5	5~10
Bow	ranged	1~2	2~3	10~15
Crossbow	ranged	1~2	4~5	15~20
Trident	polearm	4~5	4~4	3~4
Pilum	polearm	3~4	3~3	2~4
Spear	polearm	2~3	2~2	2~4
Axe	melee	4~5	4~5	1~1
Sword	melee	2~3	3~4	1~1
Club	melee	2~3	1~2	1~1

Numeric feature: lower_bound ~ upper_bound
Symbolic feature: value_1/.../value_n
Both use uniform distribution

Created based on our personal knowledge, Google image, Wikipedia ...

The Hunting Domain Data

Name	Health	Attack	Damage	Defense	Aggressivi	Swiftness	Speed
elephant	5~5	5~5	5~5	4~4	3~3	2~2	3~3
rhino	5~5	5~5	5~5	5~5	4~4	2~2	3~3
tiger	4~4	5~5	5~5	3~3	5~5	4~4	4~4
bear	4~4	5~5	5~5	4~4	5~5	3~3	3~3
wolf	3~3	4~4	4~4	3~3	3~3	4~4	4~4
badger	3~3	3~3	3~3	3~3	4~4	3~3	3~3
tortoise	1~1	2~2	2~2	5~5	1~1	1~1	1~1
armadillo	2~2	2~2	2~2	5~5	1~1	1~1	1~1
deer	2~2	2~2	2~2	2~2	2~2	4~4	4~4
sheep	2~2	2~2	2~2	2~2	2~2	3~3	3~3
antelope	1~1	1~1	1~1	1~1	1~1	5~5	5~5
rabbit	1~1	1~1	1~1	1~1	1~1	5~5	3~3

Animal functional features
“Expensive” to observe
Small variance

Name	NOSE-TYPE	BODY-SHAPE	has-horn	has-tusk	color	SIZE	leg-ratio	tail-ratio	motion-agility
elephant	long	bulky	no	yes/no	gray/dark	4~5	3~5	2~4	0.5~2
rhino	extrude	bulky	yes	no	white/gray	4~4.5	2~3	2~3	0.5~2
tiger	flat	long	no	no	striped-yellow-black	2.5~3.5	2~4	4~5	1~4
bear	extrude	bulky/fit	no	no	black/gray/brown/white	3~4	2~3	1~1	0.5~3
wolf	extrude	fit	no	no	white/gray	2~2.5	3~4	3~4	1~4
badger	extrude	flat	no	no	stripped-gray-white	1~2	1~2	2~3	3~4
tortoise	flat	plate	no	no	green/gray	0.5~1.5	1~2	2~3	0~1
armadillo	pointed	round	no	no	brown/gray	0.5~2.5	1~2	3~5	0.5~3
deer	extrude	bulky/fit/slim	yes/no	no	brown/gray	2~3	4~5	1~2	3~5
sheep	extrude	chubby/fit	yes/no	no	white/gray/brown/black	1.5~2.5	3~4	1~2	1~4
antelope	extrude	fit/slim	yes/no	no	brown/white	1~2.5	4~5	1~2	4~5
rabbit	flat	chubby/fit	no	no	gray/mixed/white/brown	0.5~1.5	1~3	1~3	3~5

Animal predictive features
“Cheap” to observe
Large variance

The Hunting Domain Data

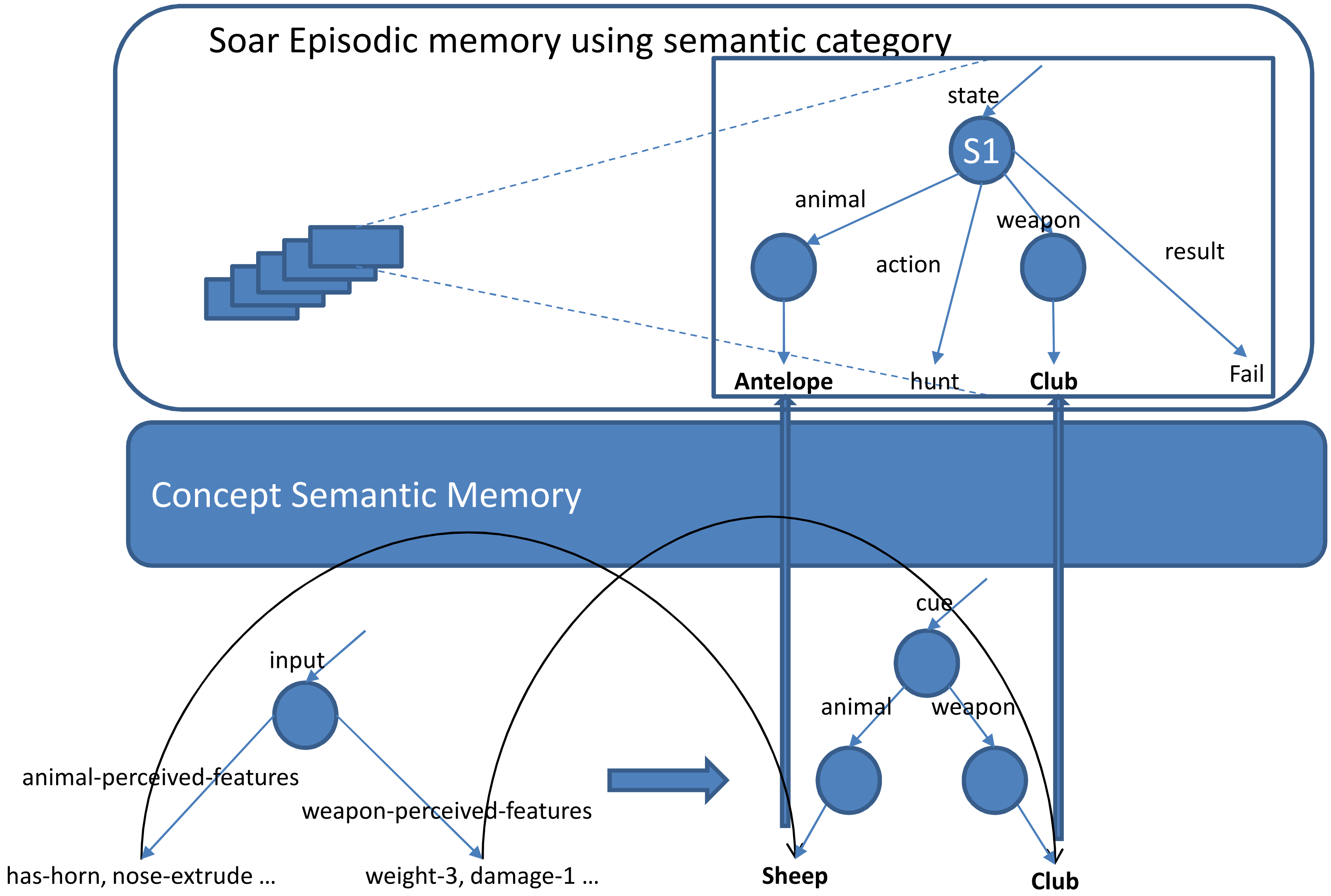
Outcomes of interactions (hunting animal with weapon)
success – 1, failure – 0
Completely deterministic – for simplicity

	Sling	Bow	Crossbow	Trident	Pilum	Spear	Axe	Sword	Club
elephant	0	0	0	0	0	0	1	0	0
rhino	0	0	0	0	0	0	1	0	0
tiger	0	0	1	1	1	1	0	0	0
bear	0	0	0	1	1	1	1	0	0
wolf	0	1	1	1	1	1	0	1	1
badger	1	1	1	1	1	1	0	1	1
tortoise	0	0	0	0	0	0	1	1	0
armadillo	0	0	0	0	0	0	1	1	1
deer	1	1	1	0	0	0	0	0	0
sheep	1	1	1	0	0	0	0	0	0
antelope	1	1	1	0	0	0	0	0	0
rabbit	1	1	1	0	0	0	0	0	0

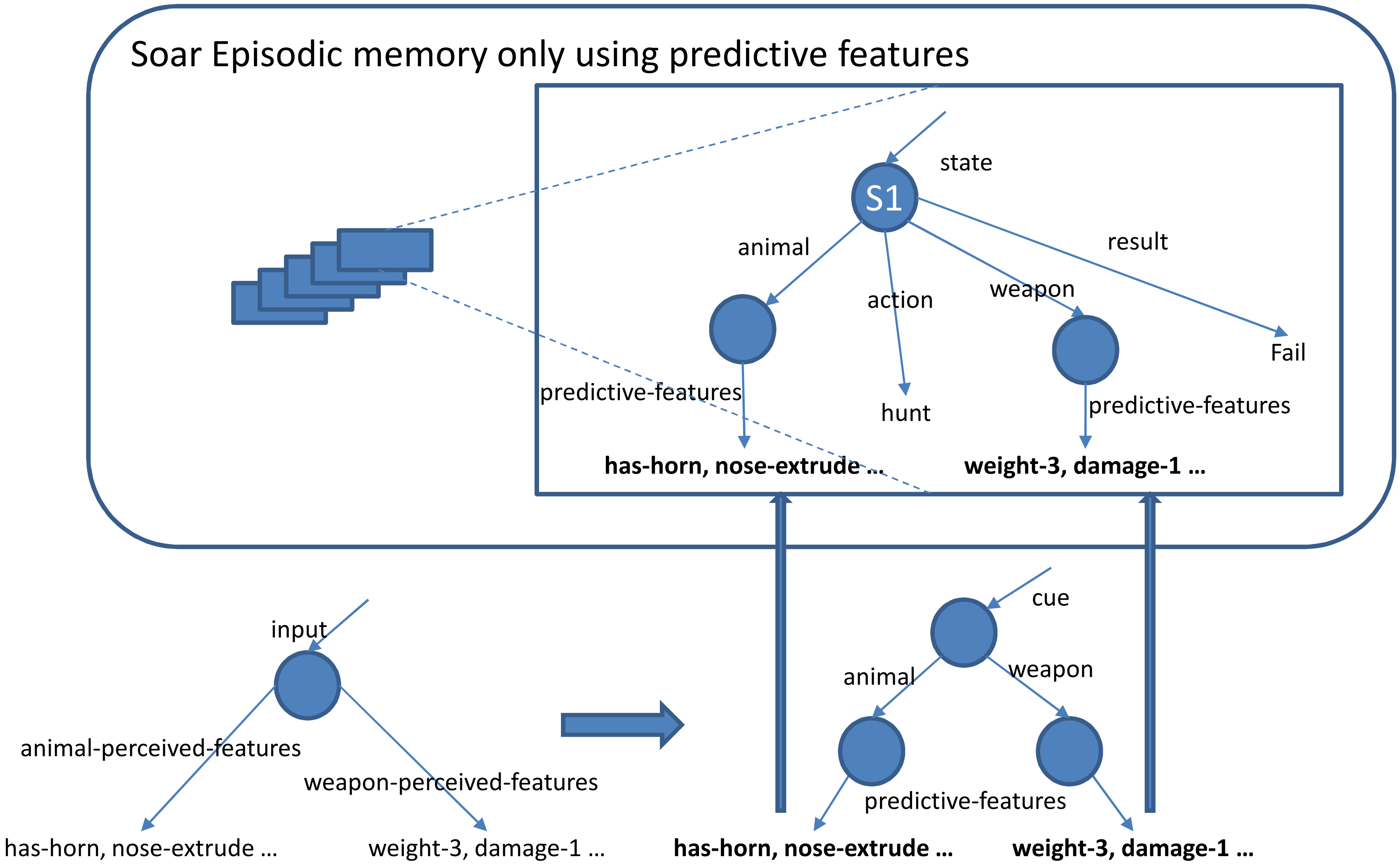
Soar Episodic Memory Approach

- Training
  - The agent is presented with specific animal and weapon
  - Perform an action (hunt) and observes the result
  - Record the experience in episodic memory
- Testing
  - Given a specific weapon and animal, retrieve the “best match” episode
  - Use the retrieved episode to predict the potential result and make the decision

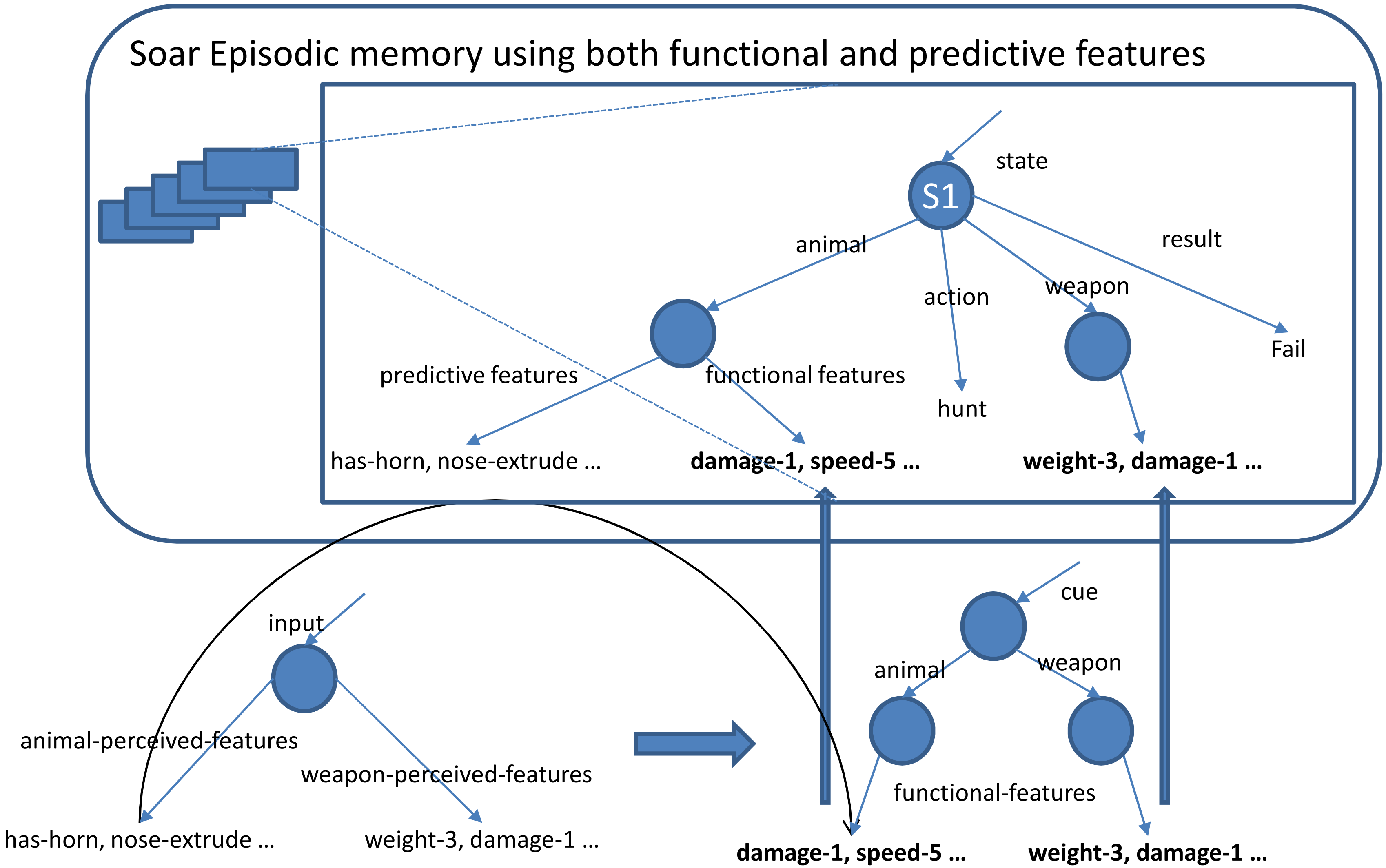
Condition 1


Soar Episodic memory using semantic category
state
S1
animal
action
weapon
result
Antelope
hunt
Club
Fail
Concept Semantic Memory
cue
input
animal
weapon
animal-perceived-features
weapon-perceived-features
has-horn, nose-extrude …
weight-3, damage-1 …
Sheep
Club

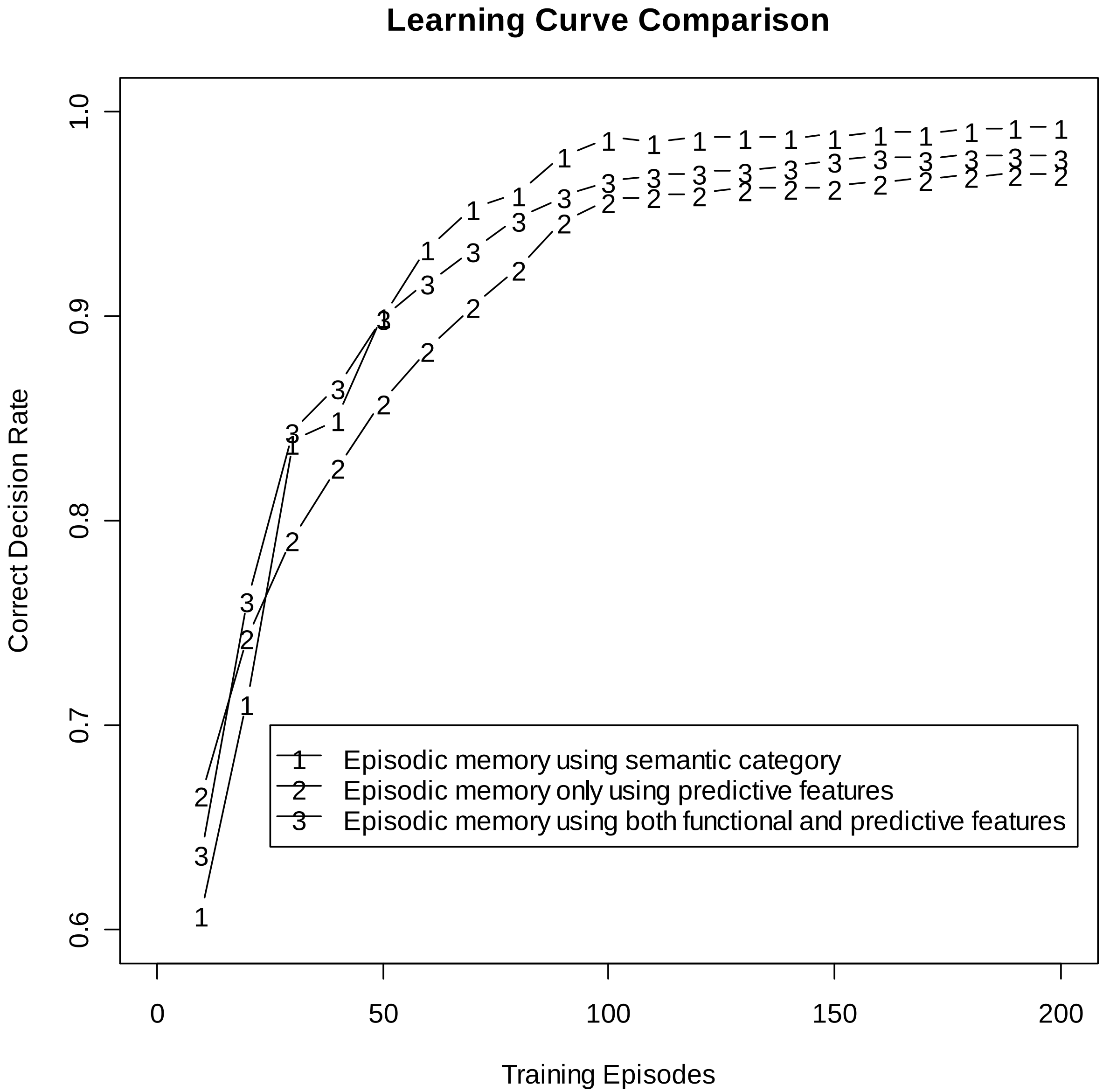
Condition 2


Soar Episodic memory only using predictive features
state
S1
animal
action
weapon
result
Fail
predictive-features
hunt
predictive-features
has-horn, nose-extrude …
weight-3, damage-1 …
input
cue
animal
weapon
animal-perceived-features
weapon-perceived-features
predictive-features
has-horn, nose-extrude …
weight-3, damage-1 …
has-horn, nose-extrude …
weight-3, damage-1 …

Condition 3



Soar Episodic memory using both functional and predictive features
state
S1
animal
result
action
weapon
Fail
predictive features
functional features
hunt
has-horn, nose-extrude …
damage-1, speed-5 …
weight-3, damage-1 …
input
cue
animal
weapon
animal-perceived-features
weapon-perceived-features
functional-features
has-horn, nose-extrude …
weight-3, damage-1 …
damage-1, speed-5 …
weight-3, damage-1 …


Learning Curve Comparison
Correct Decision Rate
Training Episodes
0.6
0.7
0.8
0.9
1.0
0
50
100
150
200
1 Episodic memory using semantic category
2 Episodic memory only using predictive features
3 Episodic memory using both functional and predictive features

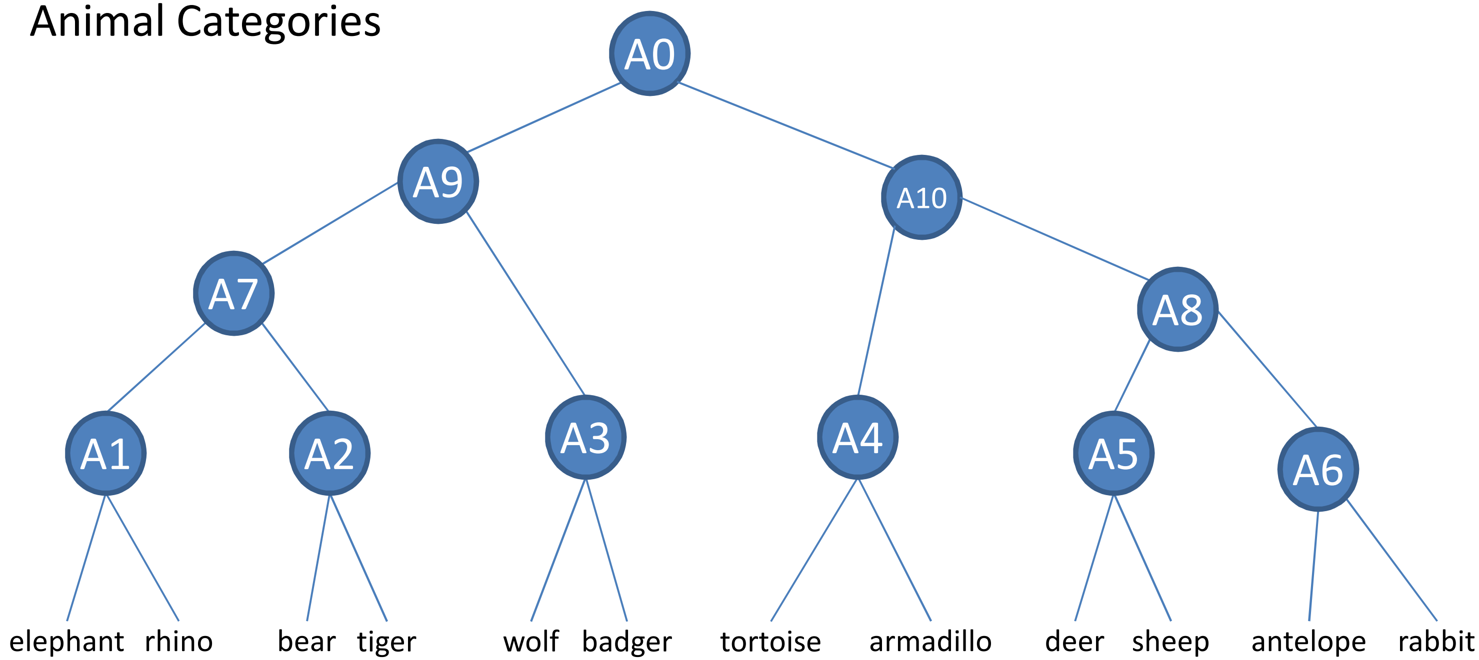
Conclusions from the plot

- Specific functional features are more reliable than general perceptual features
- Instance based learning is faster in the beginning, because probabilistic learning must accumulate enough samples
- After sufficient sampling, probabilistic learning outperforms, because it assumes a more compact model

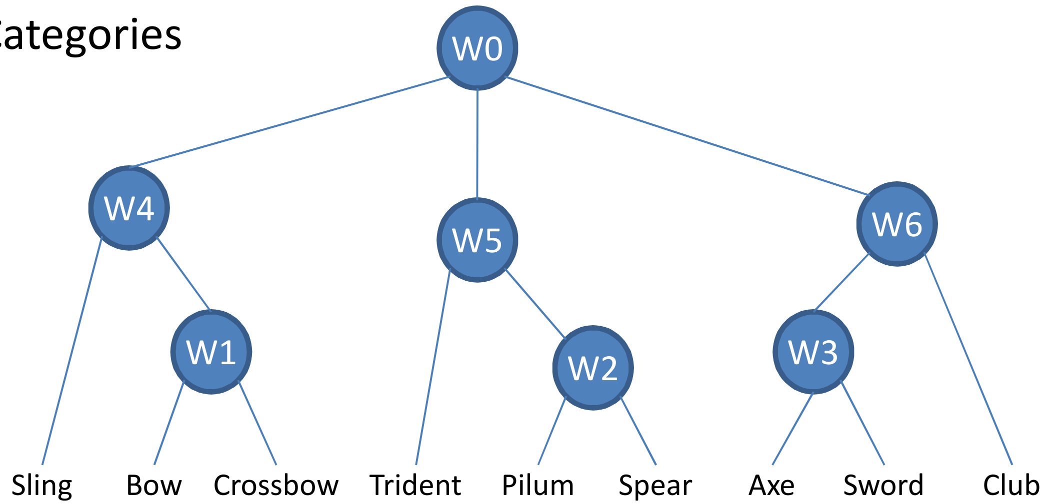
Soar-RL Approach

- Training
  - The agent is presented with specific animal and weapon
  - Perform an action (hunt/avoid) and receives a reward
  - Update numeric preference based on rewards
- Testing
  - Choose the action with the highest numeric preference (Q-value)

Animal Categories



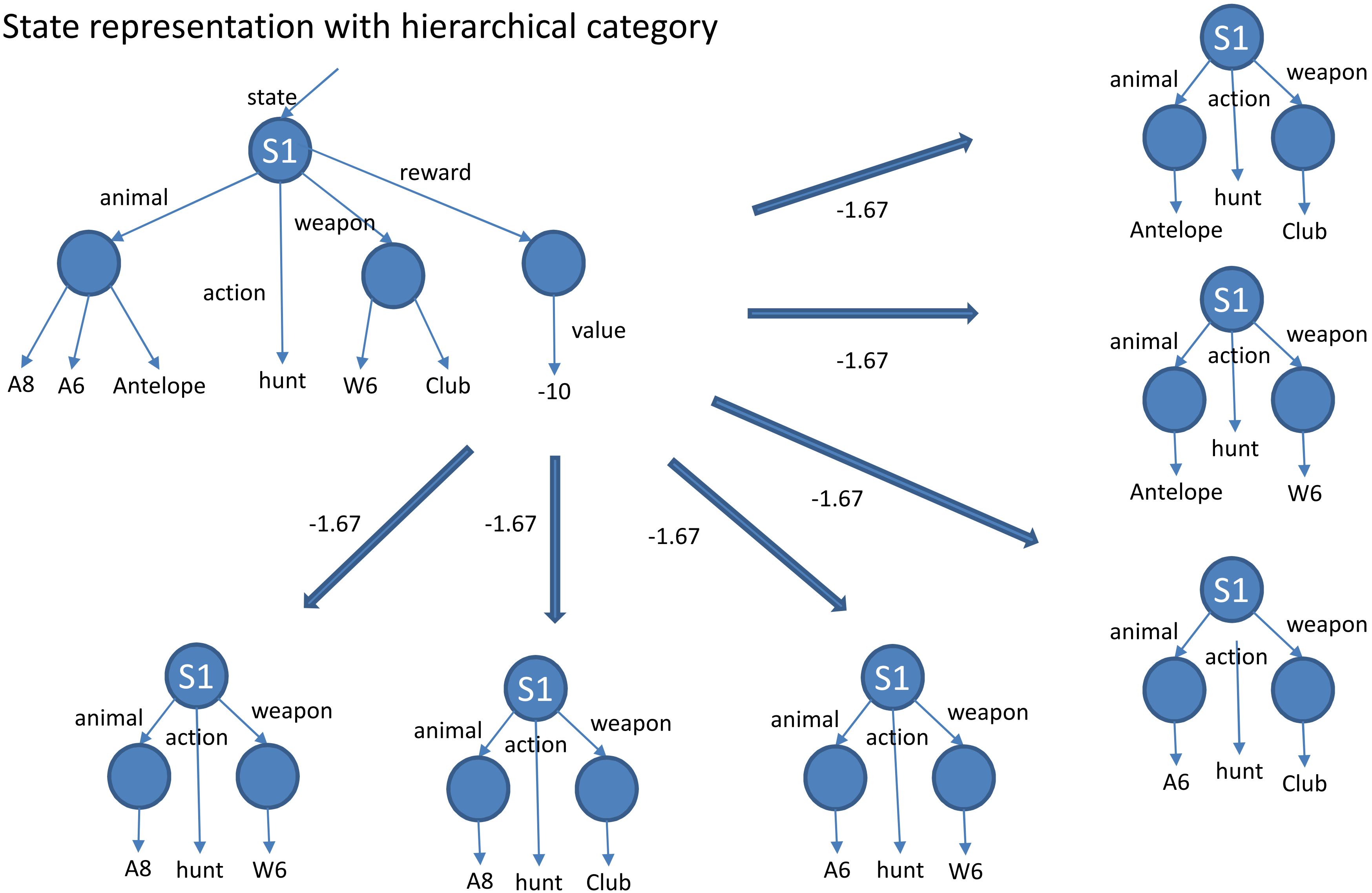
Weapon Categories



Antelope + Club + hunt = Failure

Individual RL rules

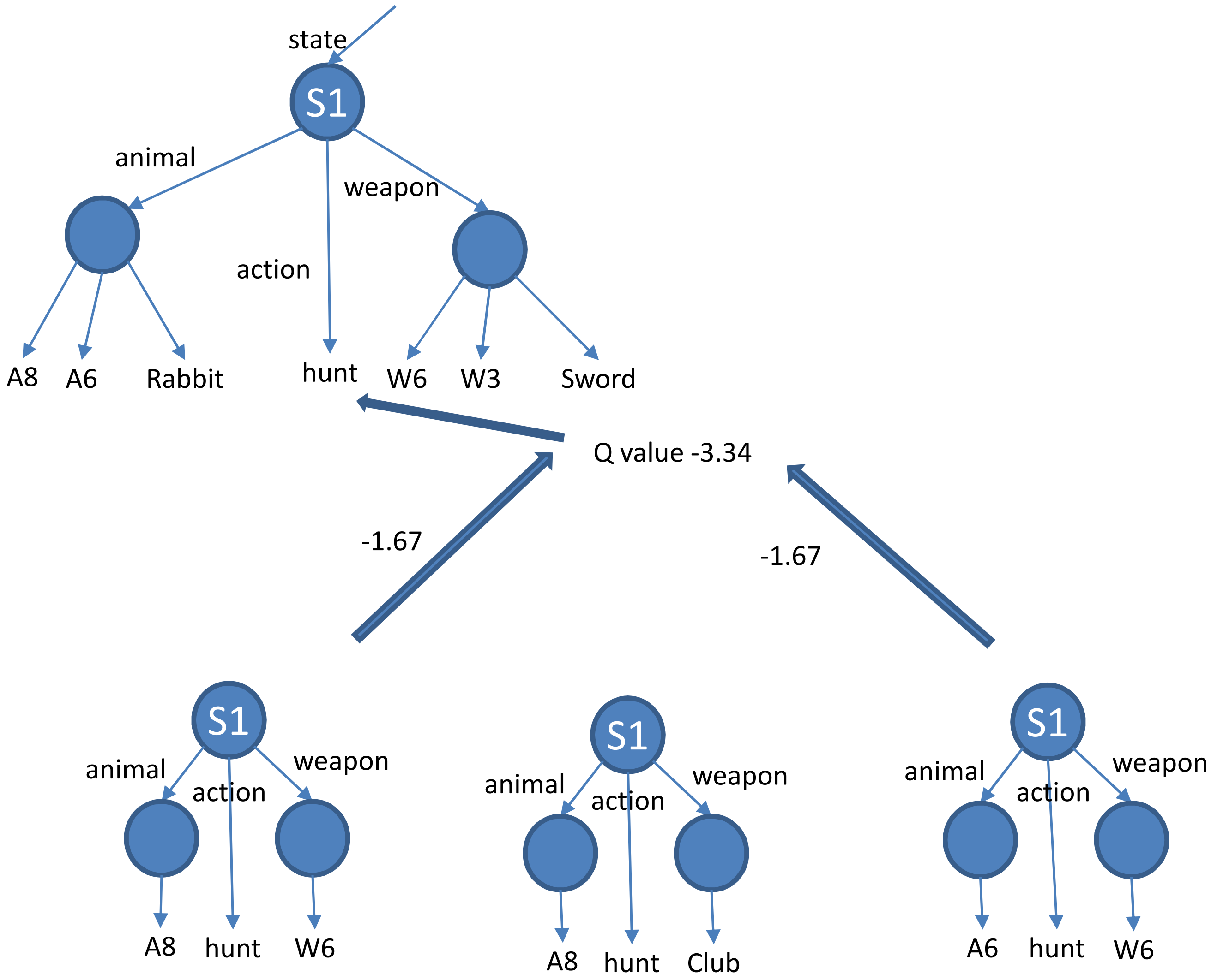
State representation with hierarchical category



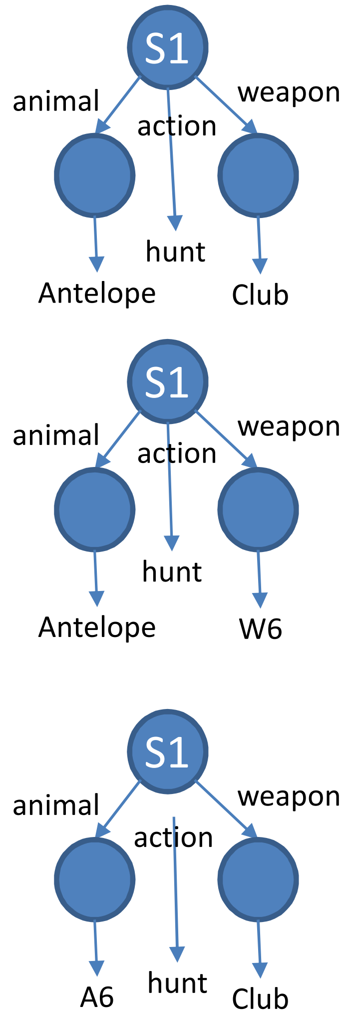
Individual RL rules

Rabbit + Sword + hunt = ?

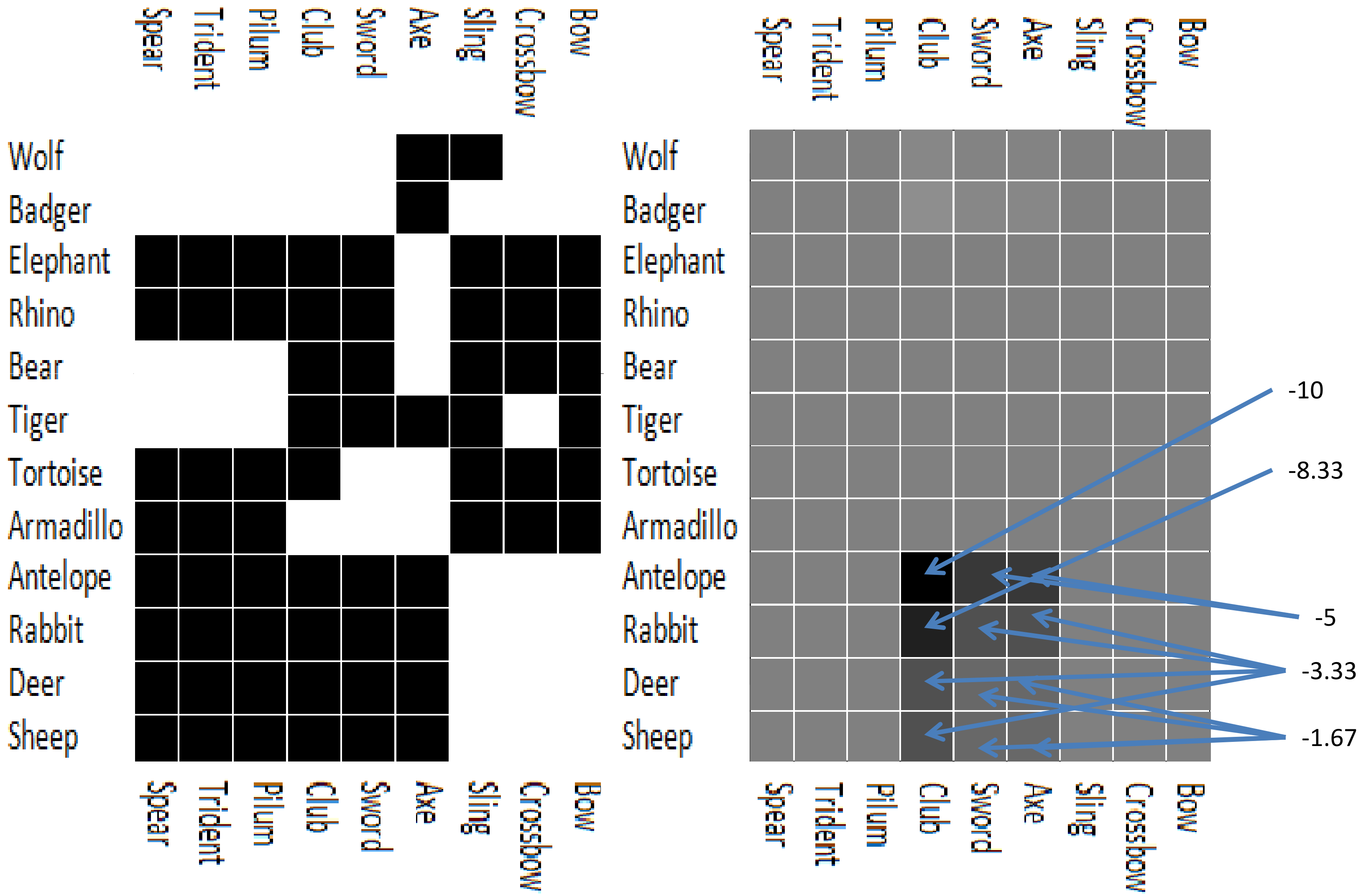
State representation with hierarchical category

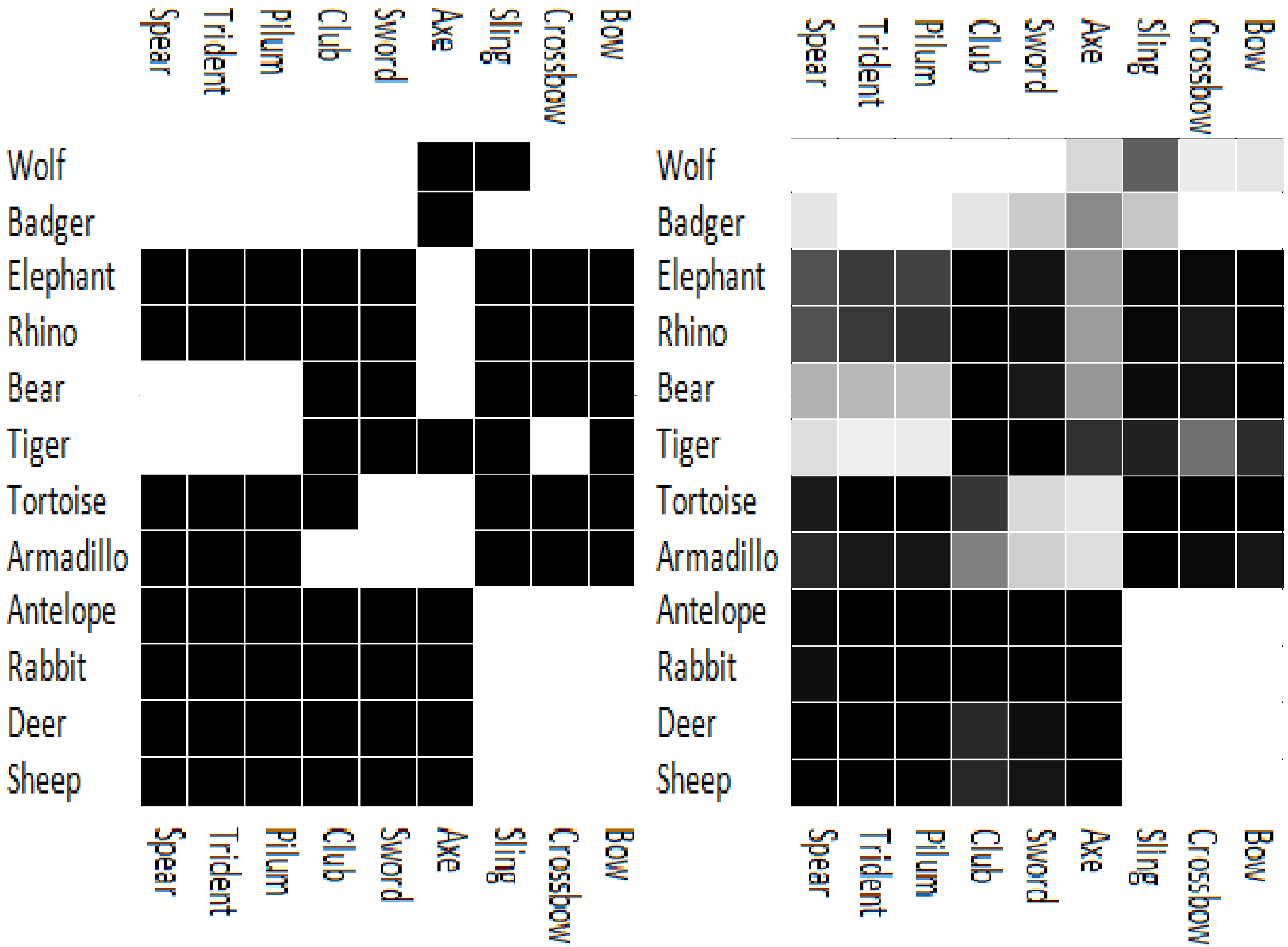


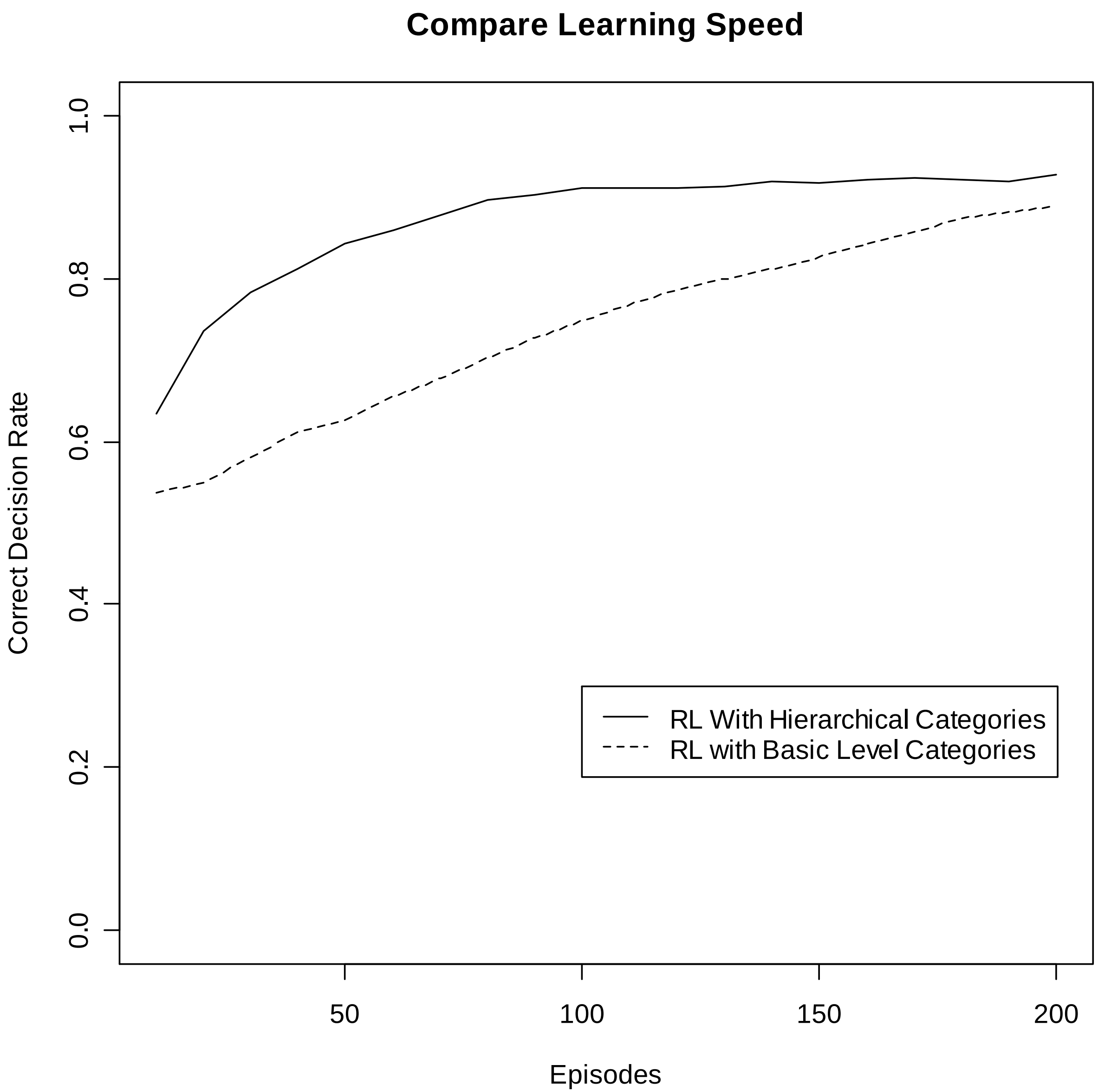
Individual RL rules



Individual RL rules


Spear
Trident
Pilum
Club
Sword
Axe
Sling
Crossbow
Bow
Wolf
Badger
Elephant
Rhino
Bear
Tiger
Tortoise
Armadillo
Antelope
Rabbit
Deer
Sheep
-10
-8.33
-5
-3.33
-1.67


Spear
Trident
Pilum
Club
Sword
Axe
Sling
Crossbow
Bow
Wolf
Badger
Elephant
Rhino
Bear
Tiger
Tortoise
Armadillo
Antelope
Rabbit
Deer
Sheep
Spear
Trident
Pilum
Club
Sword
Axe
Sling
Crossbow
Bow
Wolf
Badger
Elephant
Rhino
Bear
Tiger
Tortoise
Armadillo
Antelope
Rabbit
Deer
Sheep


Compare Learning Speed
Correct Decision Rate
0.0
0.2
0.4
0.6
0.8
1.0
50
100
150
200
Episodes
RL With Hierarchical Categories
RL with Basic Level Categories

Conclusions from the plot

- Integration of category learning helps RL with better generalization
- RL with hierarchical category representation converges slower on the horizon (not shown)
  - “Wrong generalization” is always the tradeoff
  - In the worst case, some specific situations may NEVER be learned correctly
- A possible solution is to “switch” to using only most specific rules after certain point
  - It can be made architectural
  - Similar, in spirit, to the idea of decaying learning rate and exploration rate in standard RL

Nuggets and Coal

- Nuggets
  - Added Architectural Concept Semantic Memory
    - Sub-symbolic probabilistic category learning
  - First computational models for functional integration of category learning in a general cognitive architecture
    - Integrated with "Episodic Memory" and improved learning performance
    - Integrated with Soar-RL and improved learning performance
- Coal
  - Need fully integration with real Soar Episodic Memory
  - Further improvement on integration with Soar-RL
  - Evaluation scenario is still relatively simple