A Unified Framework for Learning and Processing Perceptual, Relational, and Meta Knowledge

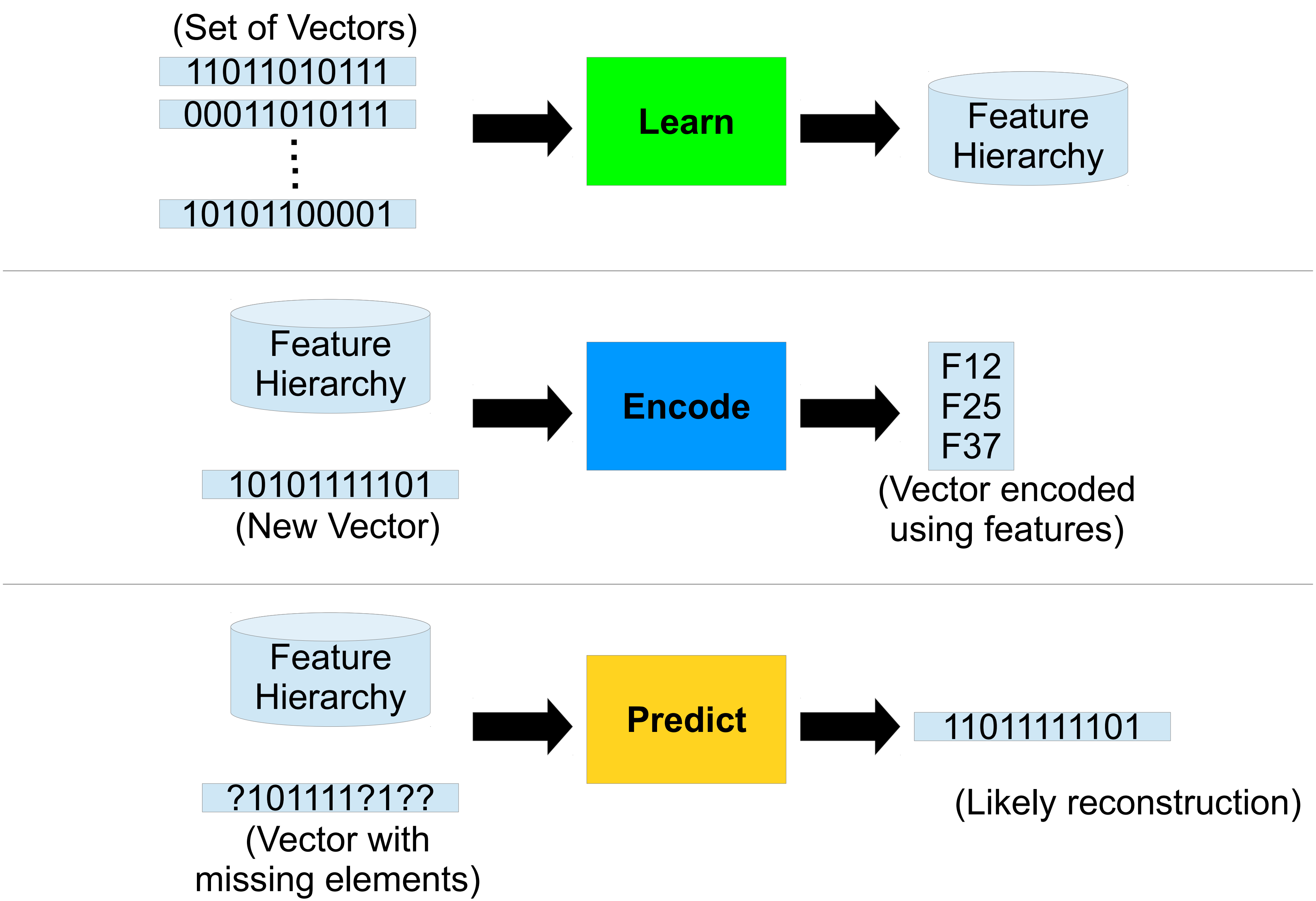
Marc Pickett

NRC/Naval Research Laboratory
Washington, DC

Maryland Metacognition Seminar
2014/01/07

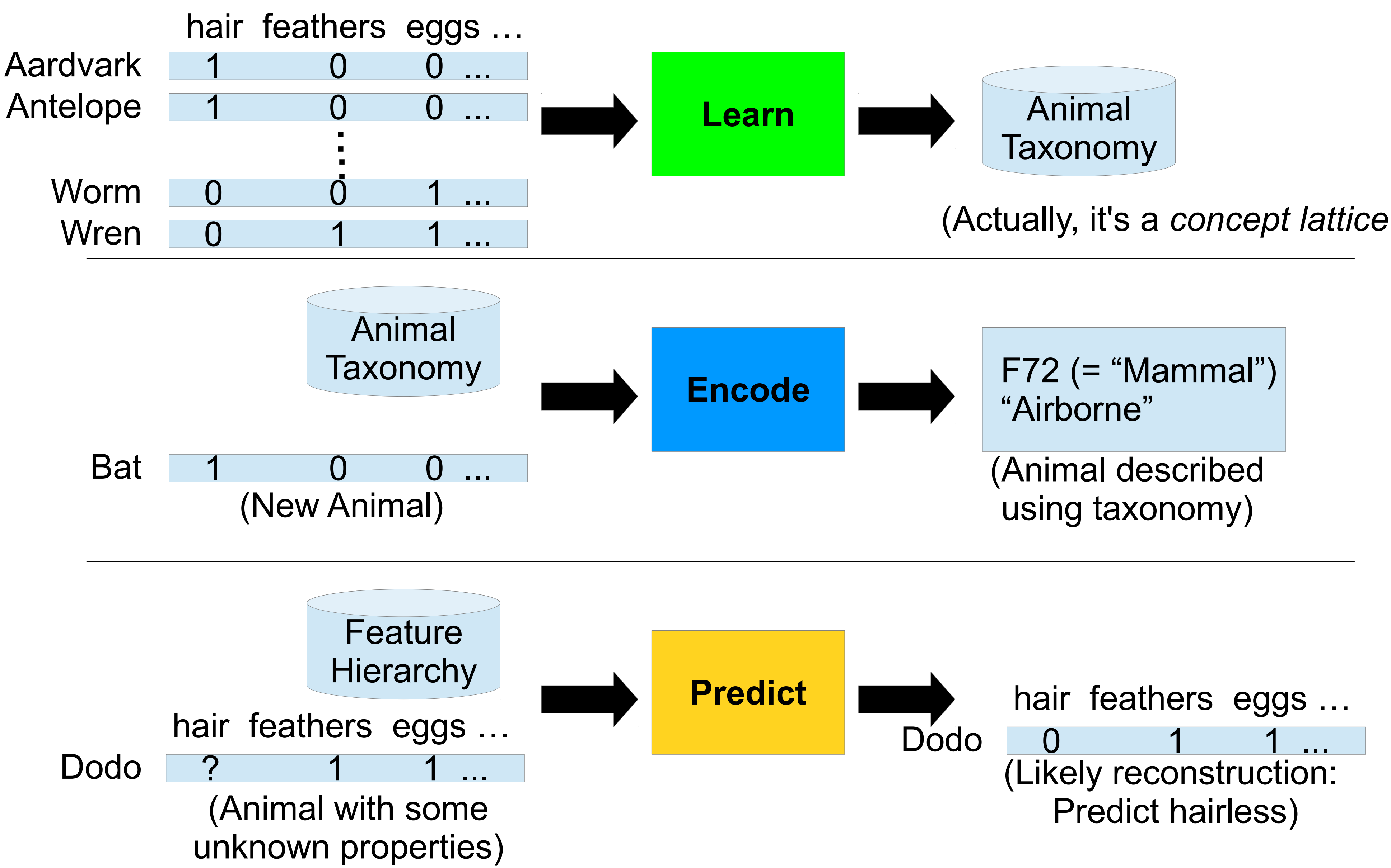
The “Vector Toolkit”

3 algorithms on **fixed-width vectors**

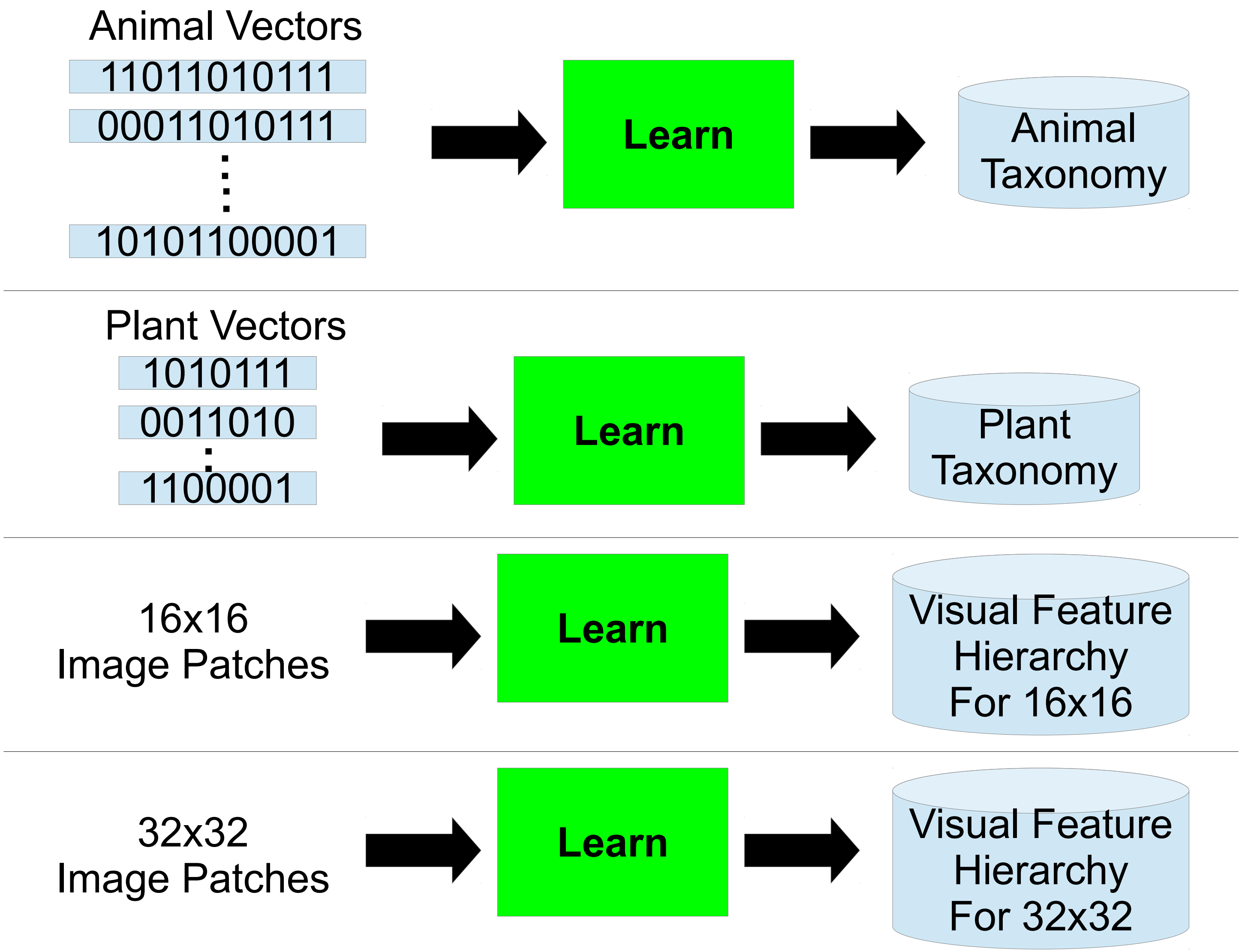


Vector Toolkit example

Animal Dataset (from UCI)



Can Learn Many Feature Hierarchies

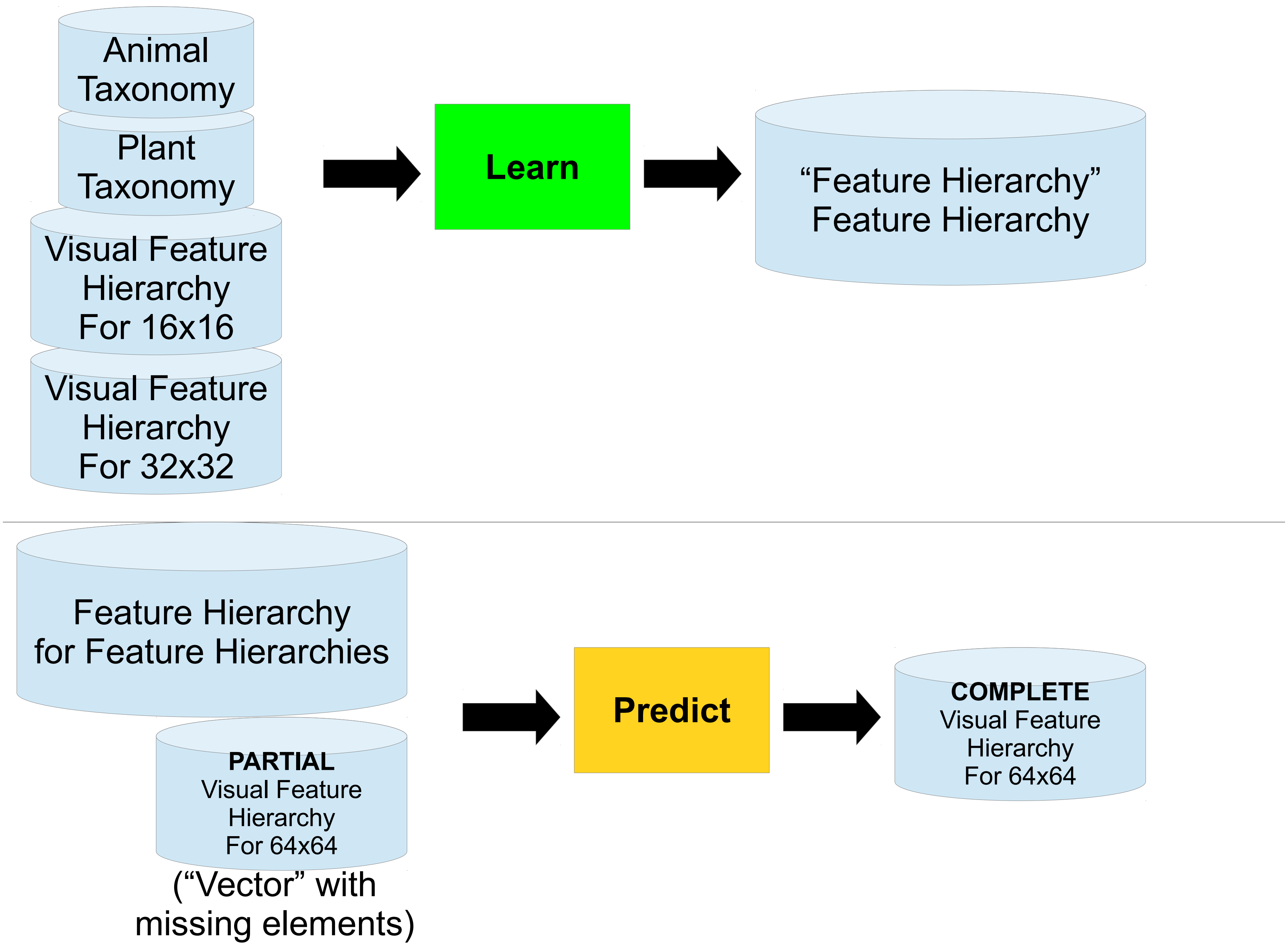


How is Animal Taxonomy like Plant Taxonomy?
Can we generalize knowledge about Image Patches?

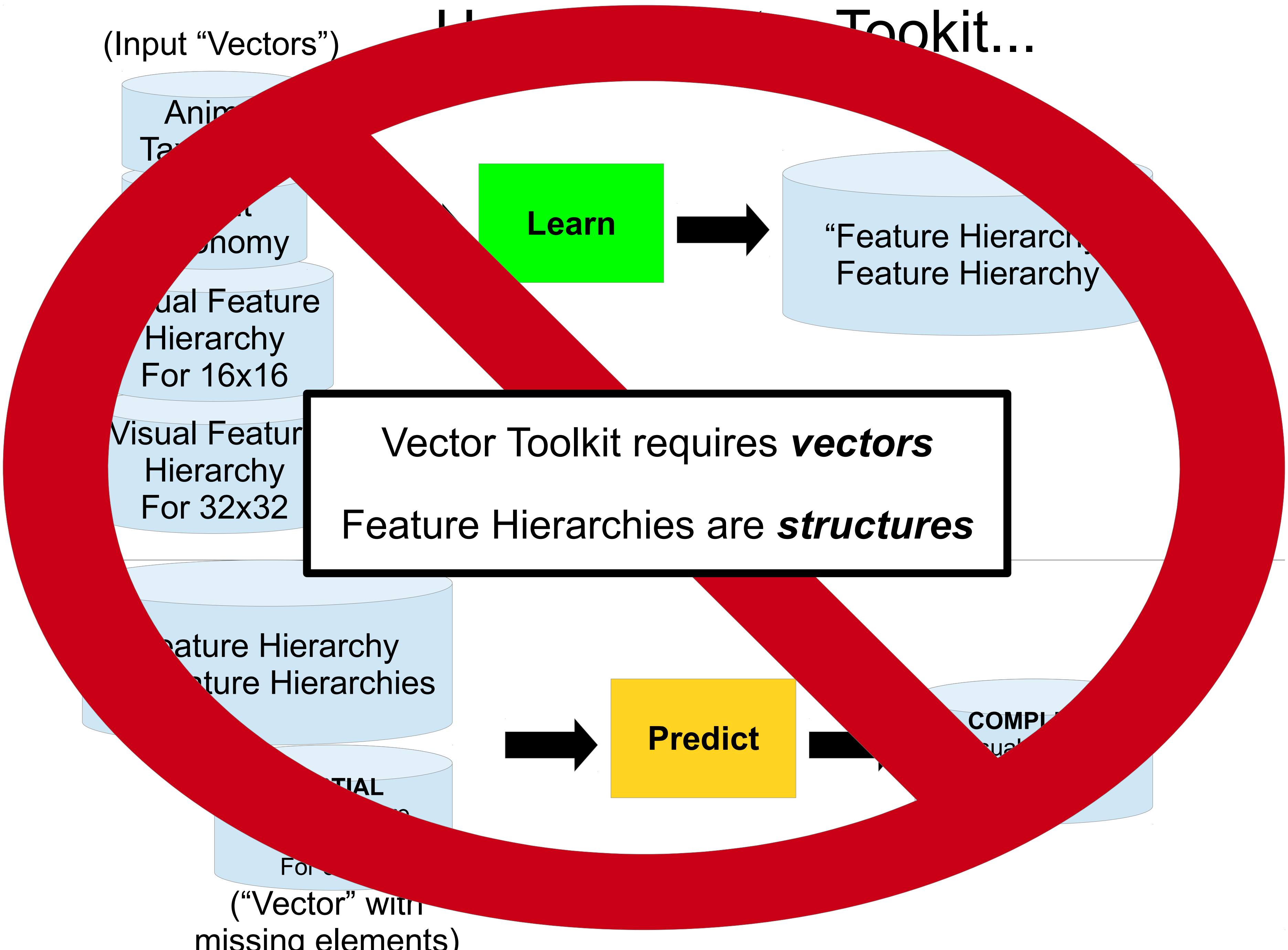
A “Meta” Feature Hierarchy?

Use the Vector Tookit...

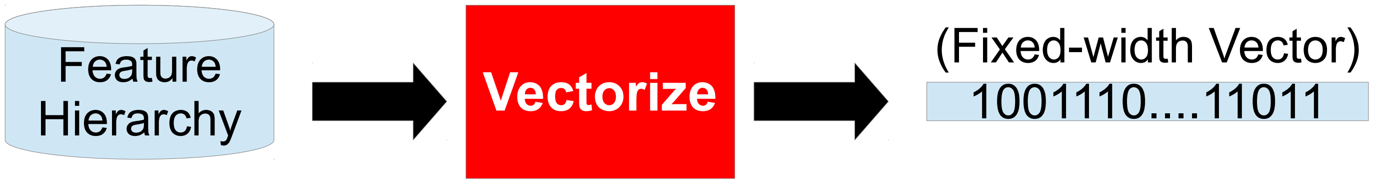
(Input “Vectors”)



A “Meta” Feature Hierarchy?



Transform Feature-Hierarchies into Vectors



Transform Feature-Hierarchy into vector such that:

Partial Overlap in vectors iff Partial *Structural* Overlap in Feature-Hierarchies

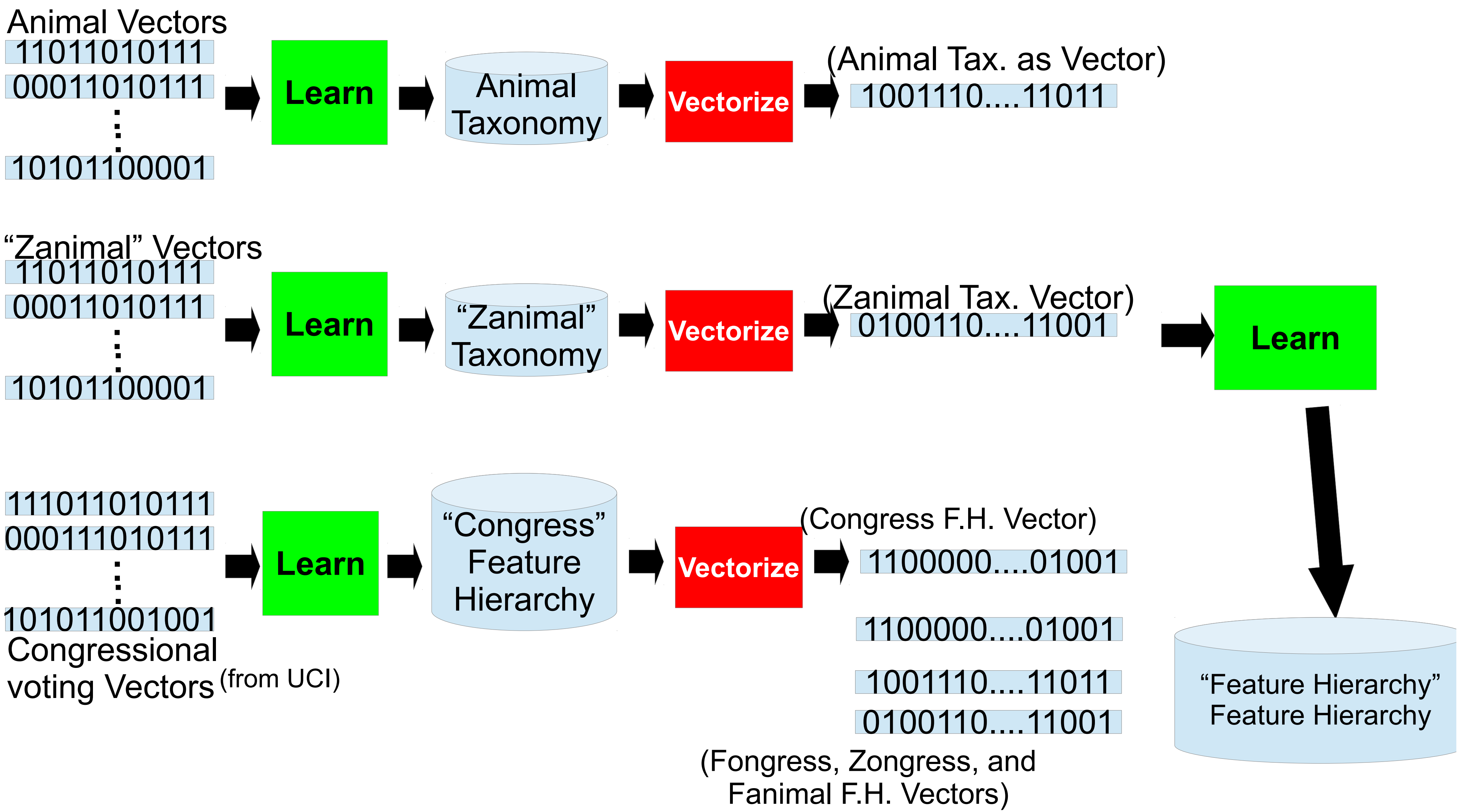
E.g., if there is a large partial isomorphism between animal taxonomy and plant taxonomy, then their vector representations will have many common elements (and vice versa).

Other approaches (Plate's HRRs, Socher's autoencoders, Bag of Words) lack this property.

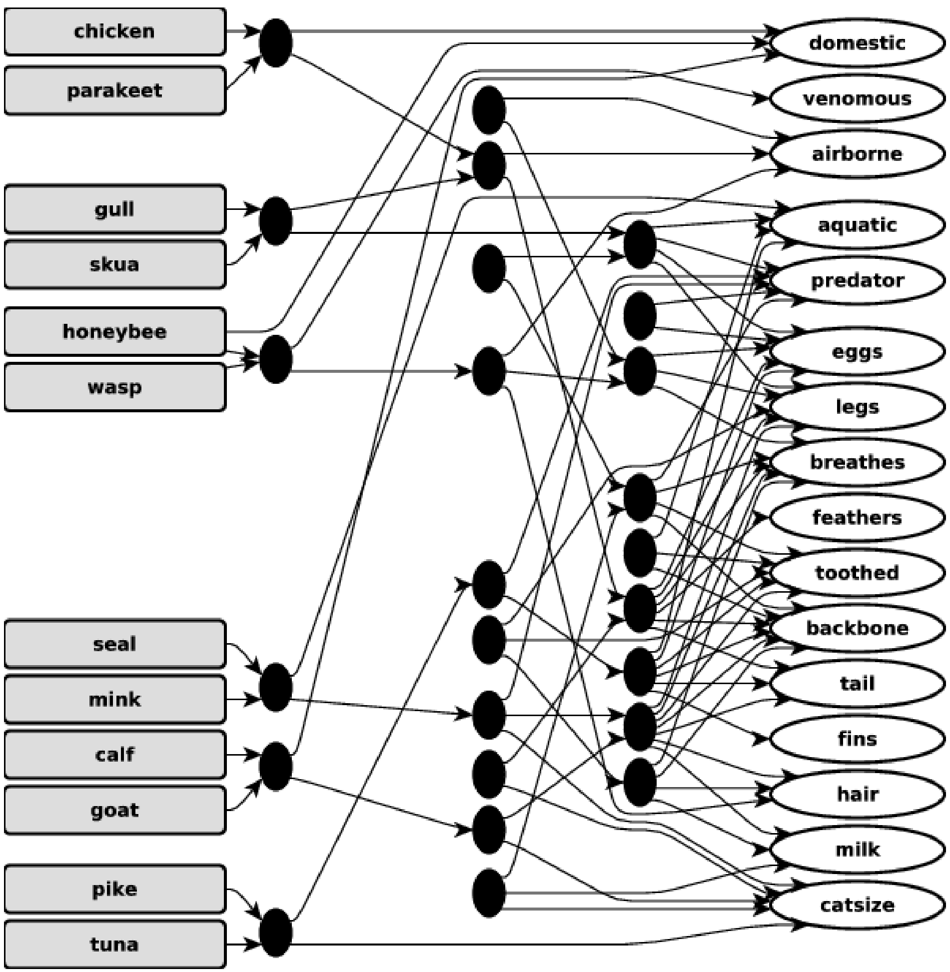
(See backup slides for details of Vectorize.)

Simple Demo

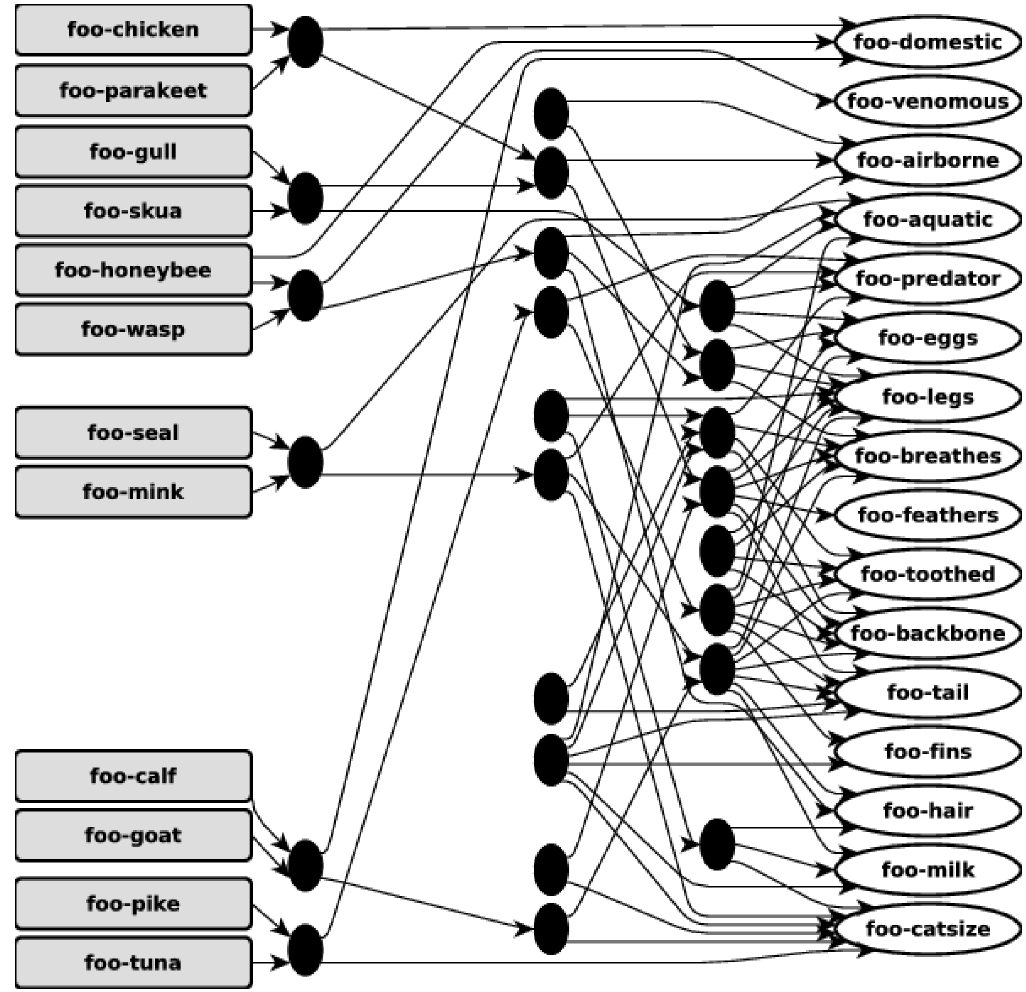
Process Finds Structure Similarity in (Object-level) Feature-Hierarchies



Simple Demo
A Peek Under The Hood

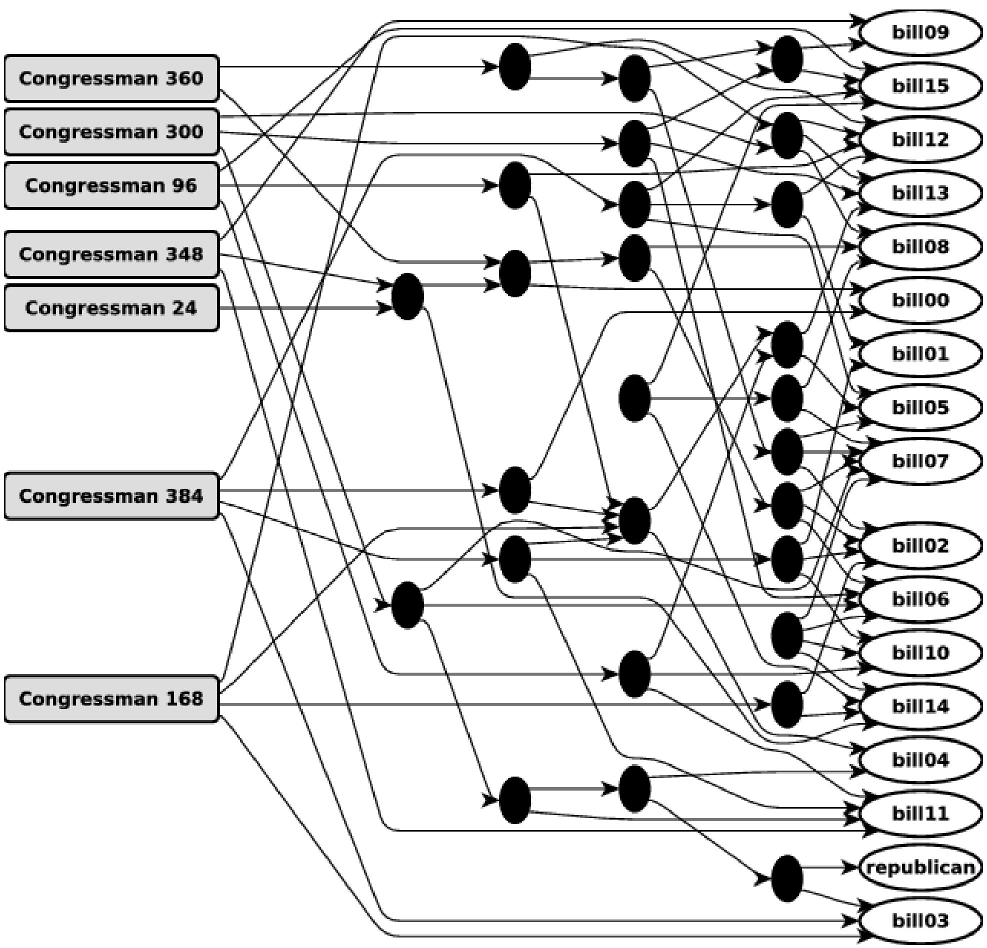


Actual Animal Taxonomy (partial)

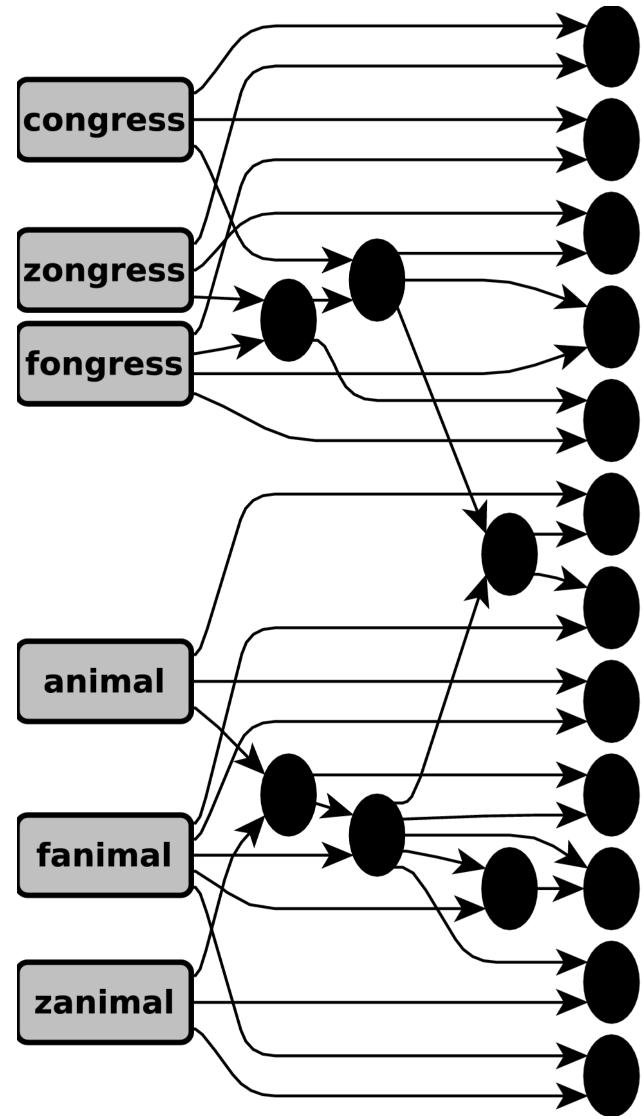


Actual Fanimal Taxonomy (partial)

Simple Demo
A Peek Under The Hood



Actual Congress Feature Hierarchy (partial)



The “Meta” Feature-Hierarchy (partial)

Discussion

Next Steps: Potential Uses of “Meta” Feature Hierarchies

- Transfer between domains
  - Learn about 64x64 image patches if 32x32 and 16x16 are already learned
  - Use **Encode** and **Predict** to make inferences for new domains
- Discover translation invariance in images
  - Feature hierarchy for top-left of image is structurally similar to bottom-right

Backup Slides

A Unified Framework for Learning and Processing Perceptual, Relational, and Meta Knowledge

Marc Pickett

NRC/Naval Research Laboratory, Washington DC

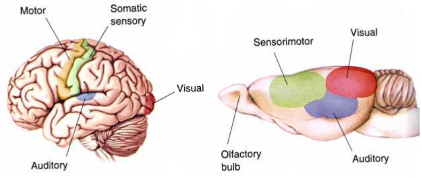
Maryland Metacognition Seminar
College Park, MD, 2014/01/07

Of Mice and Men

Minds of Mice and Men differ...

	Mice	Humans
Perception	✓	✓
Language		✓
Symbolic Reasoning		✓
Planning	?	✓
Metacognition		✓

Brains are remarkably similar...



	Mice	Humans
Cerebral Cortex	✓	✓
Thalamus	✓	✓
Hippocampus	✓	✓
Hypothalamus	✓	✓
Cerebellum	✓	✓
etc.	✓	✓

- No special structures in (newborn) human brains
- Core diffence is vastly expanded cortex in humans
- Newborn non-sensory cortex looks like sensory cortex
- **Hypothesis:** $\exists$ substrate for perception, relations, & metacognition
- **How can "cortex" be leveraged for Metacognition?**

Outline

1. Problem: How can cortex be leveraged for Metacognition?
2. Cortex: A simple "cortical" model
3. Relational Transform: structures $\rightarrow$ vectors
4. Meta-ontologies: ontologies of ontologies

Outline

1. Problem: How can cortex be leveraged for Metacognition?
2. Cortex: A simple "cortical" model
3. Relational Transform: structures $\rightarrow$ vectors
4. Meta-ontologies: ontologies of ontologies

Outline

A Simple "Cortical" Model (Pickett, 2011)

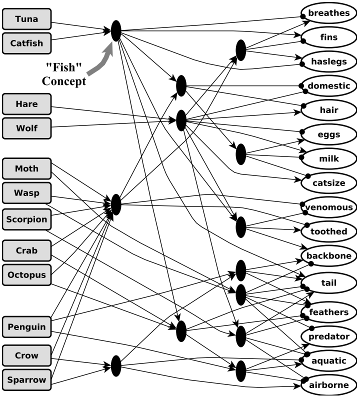
Given set of <u>uninterpreted</u> vectors, system:

- **learns** feature hierarchy (ontology) using chunking
- **parses** new instances using learned features (characterizes instance in terms of higher features)
- **predicts** missing elements using top-down inference

Animals (toy example)

Name	hair	feathers	eggs	$\cdots$
aardvark	1	0	0	$\cdots$
antelope	1	0	0	$\cdots$
bass	0	0	1	$\cdots$
bear	1	0	0	$\cdots$
		$\ddots$		
worm	0	0	1	$\cdots$
wren	0	1	1	$\cdots$

$\Rightarrow$

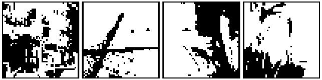


Chunking Patches from Images

Given set of <u>uninterpreted</u> vectors, system:

- **learns** feature hierarchy (ontology) using chunking
- **parses** new instances using learned features (characterizes instance in terms of higher features)
- **predicts** missing elements using top-down inference

Input: 50x50 Image Patches (uninterpreted)



$\Rightarrow$

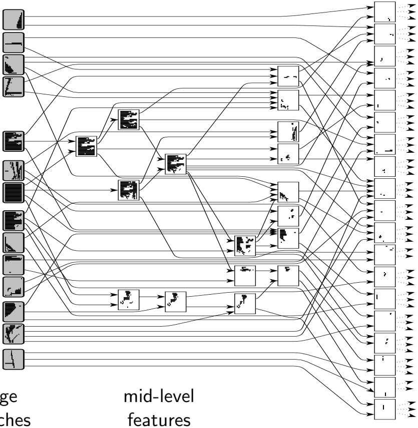


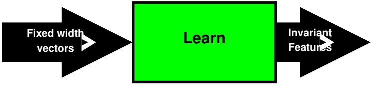
image patches

mid-level features

to raw pixels →

Hierarchy Learned from Visual Data (partial view)

What about relational data?

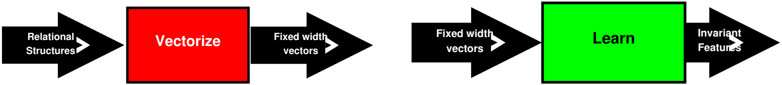


Cortical model requires fixed-width vectors...
...but much knowledge is relational.

> "A fox wanted some grapes, but could not get them. This caused him to decide that the grapes were sour, though the grapes weren't. Likewise, men often blame their failures on their circumstances, when the real reason is that they are incapable."

How does cortex represent/process relational data?

Transform Structures to Vectors



Transform relational structures to vectors.

Need to ensure <u>surface overlap</u> in vectors corresponds to <u>analogical overlap</u> in original structures

Outline

1. Problem: How can cortex be leveraged for Metacognition?
2. Cortex: A simple “cortical” model
3. Relational Transform: structures $\rightarrow$ vectors
4. Meta-ontologies: ontologies of ontologies

Structure Transformer: I

English (for clarity)

"A fox wanted some grapes, but could not get them. This caused him to decide that the grapes were sour, though the grapes weren't. Likewise, men often blame their failures on their circumstances, when the real reason is that they are incapable."

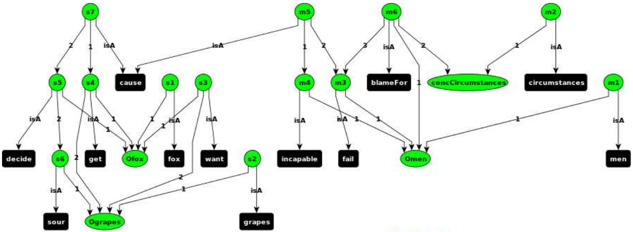
Predicate Form (actual input) from Thagard, 1990

`fox OFox`	`cause m4 m3`	`sameAs f6 (sour OGrapes)`
`false f6`	`grapes OGrapes`	`sameAs f5 (decide OFox f6)`
`cause f4 f5`	`incapable OMen`	`sameAs f4 (get OFox OGrapes)`
`false f4`	`decide OFox f6`	`sameAs m4 (incapable OMen)`
`men OMen`	`sameAs m3 (fail OMen)`	`blameFor OMen concCircum m3`
`fail OMen`	`want OFox OGrapes`	`circumstances concCircum`

Structure Transformer: I

Predicate Form (actual input) from Thagard, 1990

`fox OFox`	`cause m4 m3`	`sameAs f6 (sour OGrapes)`
`false f6`	`grapes OGrapes`	`sameAs f5 (decide OFox f6)`
`cause f4 f5`	`incapable OMen`	`sameAs f4 (get OFox OGrapes)`
`false f4`	`decide OFox f6`	`sameAs m4 (incapable OMen)`
`men OMen`	`sameAs m3 (fail OMen)`	`blameFor OMen concCircum m3`
`fail OMen`	`want OFox OGrapes`	`circumstances concCircum`

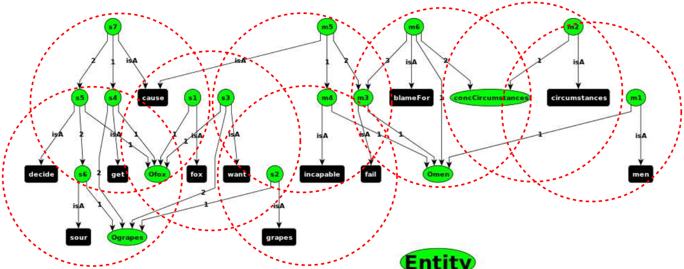


Sour Grapes Fable

Structure Transformer: I

Predicate Form (actual input) from Thagard, 1990

fox OFox	cause m4 m3	sameAs f6 (sour OGrapes)
false f6	grapes OGrapes	sameAs f5 (decide OFox f6)
cause f4 f5	incapable OMen	sameAs f4 (get OFox OGrapes)
false f4	decide OFox f6	sameAs m4 (incapable OMen)
men OMen	sameAs m3 (fail OMen)	blameFor OMen concCircum m3
fail OMen	want OFox OGrapes	circumstances concCircum



Sour Grapes Fable

Structure Transformer: I

Predicate Form (actual input) from Thagard, 1990

fox OFox	cause m4 m3	sameAs f6 (sour OGrapes)
false f6	grapes OGrapes	sameAs f5 (decide OFox f6)
cause f4 f5	incapable OMen	sameAs f4 (get OFox OGrapes)
false f4	decide OFox f6	sameAs m4 (incapable OMen)
men OMen	sameAs m3 (fail OMen)	blameFor OMen concCircum m3
fail OMen	want OFox OGrapes	circumstances concCircum

Transforming a Window

```
blameFor OMen concCircum m3
sameAs m3 (fail OMen)
fail OMen
circumstances concCircum
men OMen
incapable OMen
```

$\Longrightarrow$

```
blameFor1=blameFor3.fail1
circumstances1=blameFor2
fail1=blameFor3.fail1
fail1=blameFor1
incapable1=blameFor3.fail1
incapable1=blameFor1
incapable1=fail1
men1=blameFor3.fail1
men1=blameFor1
men1=fail1
men1=incapable1
```

Structure Transformer: II

Many Transformed Windows

```
blameFor1=blameFor3.fail1
circumstances1=blameFor2
fail1=blameFor3.fail1
fail1=blameFor1
incapable1=blameFor3.fail1
incapable1=blameFor1
incapable1=fail1
men1=blameFor3.fail1
men1=blameFor1
men1=fail1
men1=incapable1
```

```
cause2.fail1=blameFor3.fail1
blameFor1=blameFor3.fail1
blameFor1=cause2.fail1
cause2=blameFor3
fail1=blameFor3.fail1
fail1=cause2.fail1
fail1=blameFor1
men1=blameFor3.fail1
men1=cause2.fail1
men1=blameFor1
men1=fail1
```

```
false1.sour1=decide2.sour1
decide1=cause2.decide1
decide2=cause2.decide2
false1=cause2.decide2
false1=decide2
```

```
blameFor1=blameFor3.fail1
fail1=blameFor3.fail1
fail1=blameFor1
incapable1=blameFor3.fail1
incapable1=blameFor1
incapable1=fail1
men1=blameFor3.fail1
men1=blameFor1
men1=fail1
men1=incapable1
```

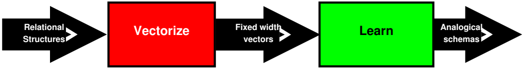
⋮

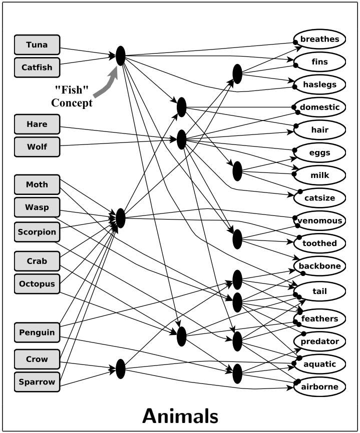
⋮

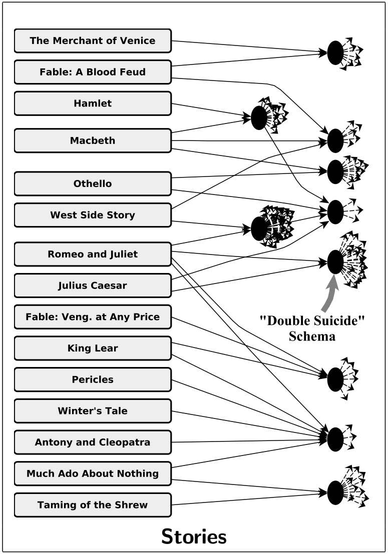
Algorithm:

- Treat transformed windows as percepts: Chunk
- Treat bags of chunked and merged windows as inputs
  - Chunk these.

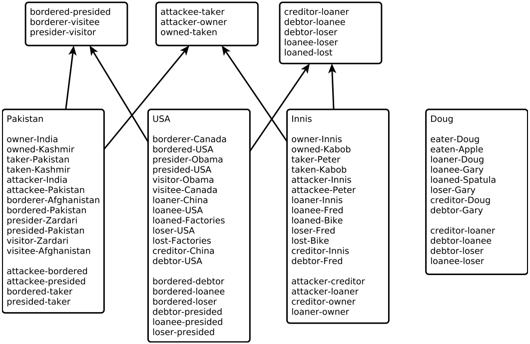
Concepts Learned from Stories



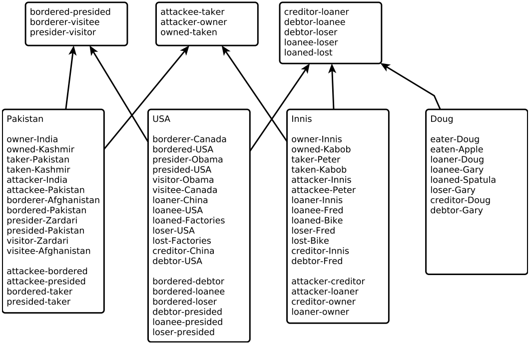




Analogical Inference with “Cortex”

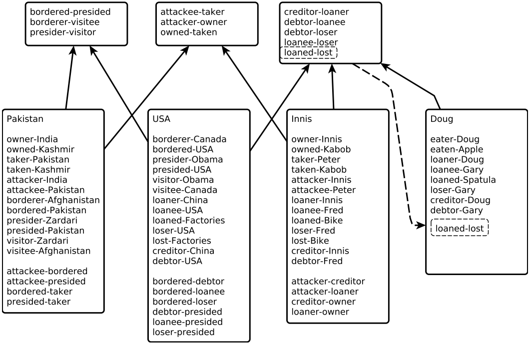


Analogical Inference with "Cortex"



- **parse** story (to inherit from top-right node)

Analogical Inference with “Cortex”



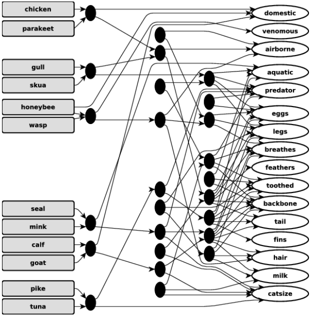
- **parse** story (to inherit from top-right node)
- **predict** (top-down) `loaned-lost` feature
- **chain** `loaned-lost` with `loaned-Spatula` to get `lost-Spatula` (i.e., the Spatula was lost)

Outline

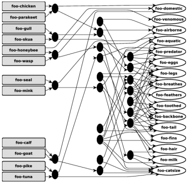
1. Problem: How can cortex be leveraged for Metacognition?
2. Cortex: A simple "cortical" model
3. Relational Transform: structures $\rightarrow$ vectors
4. Meta-ontologies: ontologies of ontologies

Outline

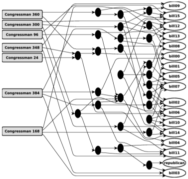
Ontologies are Relational Structures too!



The Zoo Ontology



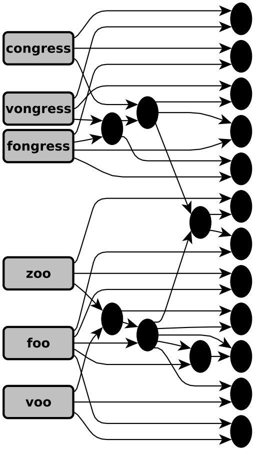
The Foo Ontology



The Congress Ontology

Ontologies as data

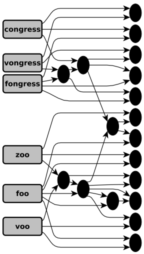
- Use relational transform
- Modify because dissimilar structures can encode similar relations (See paper for details)



An ontology of ontologies

Uses of Meta-ontologies

- Find higher-level patterns
  - Discover invariances in perceptual data (e.g., translation in vision)
  - Transfer knowledge from analogous tasks
  - Build ontology of controllers
- Metacognition
  1. Encode cognitive processes in predicate form
  2. Analyze using same machinery to analyze everything else (cortex)
  3. "Translate" back to object level



An ontology of ontologies