Harvesting Semantic Content from the Web for Higher-quality NLP

Eduard Hovy

with Zornitsa Kozareva

Information Sciences Institute
University of Southern California
www.isi.edu/~hovy

The problem with NLP today...

Webclopedia
(Hovy et al. 01)

- Where do lobsters like to live?
  — *on the table*
- Where are zebras most likely found?
  — *in the dictionary*
- How many people live in Chile?
  — *nine*
- What is an invertebrate?
  — *Dukakis*

Systems need repository of knowledge, plus ability to do commonsense reasoning

Uses for knowledge in NLP

- **Improving accuracy of IR / web search**
  TREC 98–03: recall, precision around 40%

★ Understand user query; expand query terms by meaning

- **Achieving conceptual summarization**
  Never been done yet, at non-toy level

★ Interpret topic, fuse concepts according to meaning; re-generate

- **Improving QA**
  TREC 99–04: factoids around 65%

★ Understand Q and A; match their meanings; use inference

- **Improving MT quality**
  MTEval 94: ~70%, depending on what you measure

★ Disambiguate word senses to find correct meaning

What kind(s) of knowledge would help?

- **Syntactic** information
  - Penn Treebank, Treebanks in other languages, etc.
- **Lexical semantics**
  - Framenet, WordNet, Propbank, etc.; word distributions and clusters
  - Microtheories of quantification, modality/negation, amounts, etc…
- **Temporal and spatial** information
  - TIME-ML, corpora, etc.
- **Discourse** knowledge
  - Discourse structure theories like RST, discourse corpora
- **Subjectivity**/opinion information
  - MPQA and movie opinion corpora, etc.
- **Inference** rules, entailments, and axioms
  - ?
- **Ontological** / taxonomic knowledge
  - CYC, WordNet, SUMO, Omega, etc.
- **Pragmatic** knowledge

My beliefs

- Syntactic info is useful but no longer a big problem
- Needed: Word-level semantic info, to turn terms into concepts:
  - Terms
  - Structural (frame) info associated with certain terms (like verbs)
  - ‘Definitional’ info associated with each term
  - Inter-term relations (including ISA)
- Later: More semantic and pragmatic info

- Treebanks
- Propbank, FrameNet

Missing!

Incomplete / wrong!

- WordNet
- MPQA, etc.

Credo and methodology

Ontologies (and even concepts) are too complex to build all in one step…

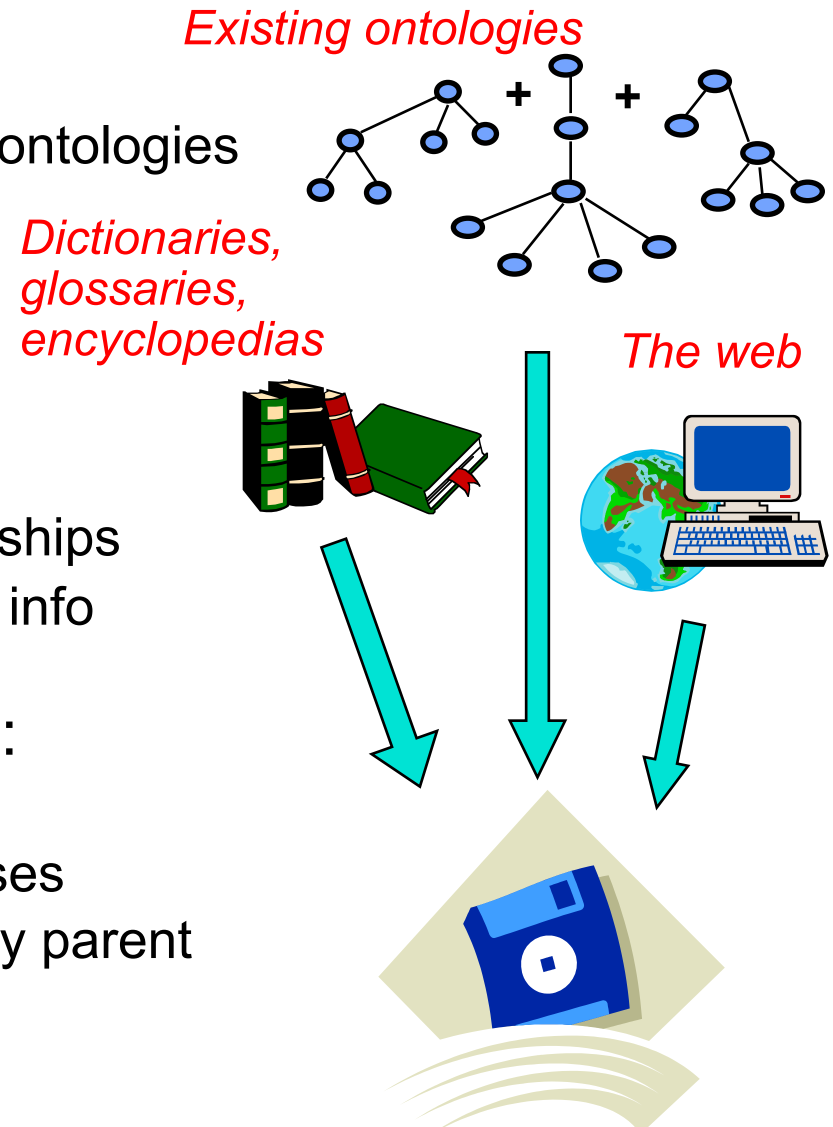
…so build them bit by bit, testing each new (kind of) addition empirically…

…and develop appropriate learning techniques for each bit, so you can automate the process…

…so next time (since there's no ultimate truth) you can build a new one more quickly

Plan: stepwise accretion of knowledge

- Initial Upper Model framework:
  - Start with existing (terminological) ontologies as pre-metadata
  - Weave them together
- Build Middle Model concepts:
  - Define/extract concept ‘cores’
  - Extract/learn inter-concept relationships
  - Extract/learn definitional and other info
- Build (large) data/instance base:
  - Extract instance ‘cores’
  - Link into ontology; store in databases
  - Extract more information, guided by parent concept



A six-step procedure

1. Starting point: **existing ontologies**
   – Cross-ontology alignment and merging

2. Converting terms to concepts
   – Term clustering and topic signatures

3. Relations and axioms
   – Harvesting relations and constraints
   – Learning axiomatic knowledge

4. Instances and Basic Level terms
   – Harvesting large numbers of instances from text

5. Intermediate terms
   – Harvesting large numbers of mid-level terms

6. Taxonomy structure
   – Organizing the mid-level terms into taxonomies

For today:

1. Starting point: **existing ontologies**
   – Cross-ontology alignment and merging
2. **Converting terms to concepts**
   – Term clustering and topic signatures
3. **Instances and Basic Level terms**
   – Harvesting large numbers of instances from text
4. **Intermediate terms / Classes**
   – Harvesting large numbers of mid-level terms
5. **Taxonomy structure**
   – Organizing the mid-level terms into taxonomies
6. **Relations and axioms**
   – Harvesting relations and constraints
   – Learning axiomatic knowledge

Part 1

CROSS-ONTOLOGY ALIGNMENT AND MERGING

Part 2

LEARNING TOPIC SIGNATURES

Topic signatures

"You know a word by the company it keeps"

Word family built around inter-word relations

- **Def**: Head word (or concept), plus set of related words (or concepts), each with strength:

$$\{ T_k, (t_{k1},w_{k1}), (t_{k2},w_{k2}), \ldots , (t_{kn},w_{kn}) \}$$

- **Problem**: Scriptal co-occurrence, etc. — how to find it?
- Approximate this by simple textual term co-occurrence...

Related words in texts show Poisson distribution:

In large set of texts, topic keywords concentrate around topics; so compare topical word frequency distributions against global background counts

Learning signatures

Procedure:

1. Collect texts, sorted by topic
2. Identify families of co-occurring words
3. Evaluate their purity
4. Find the words' concepts in the Ontology
5. Link together the concept signatures

Need texts, sorted

How to count co-occurrence?

How to evaluate?

Need disambiguator

Calculating weights

Approximate relatedness using various formulas

$$\underline{tf.idf} \; : \; w_{jk} = tf_{jk} * idf_j$$

$$\chi^2 \; : \; w_{jk} = \begin{cases} (tf_{jk} - m_{jk})^2 / m_{jk} & \text{if } tf_{jk} > m_{jk} \\ 0 & \text{otherwise} \end{cases}$$

(Hovy & Lin, 1997)

- tf_{jk} : count of term j in text k ("waiter" often only in some texts).
- $idf_j = log(N/n_j)$: within-collection frequency ("the" often in <u>all</u> texts), n_j = number of docs with term j, N = total number of documents.
- *tf.idf* is the best for IR, among 287 methods (Salton & Buckley, 1988).
- $m_{jk} = (\Sigma_j \, tf_{jk} \, \Sigma_k \, tf_{jk}) \, / \, \Sigma_{jk} \, tf_{jk}$: mean count for term j in text k .

$$\underline{\text{likelihood ratio } \lambda} : \; 2log \, \lambda = 2N \, . \, I \, (R \, ; T)$$

(Lin & Hovy, 2000)

(more approp. for sparse data; $-2log\lambda$ asymptotic to χ^2).

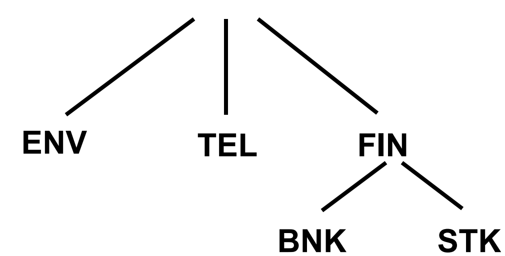
- N = total number terms in corpus.
- I = mutual information between text relevance R and given term T ,
  = $H(R) - H(R \mid T)$ for $H(R)$ = entropy of terms over relevant texts R
  and $H(R \mid T)$ = entropy of term T over rel and nonrel texts.

Early signature study

(Hovy & Lin 97)

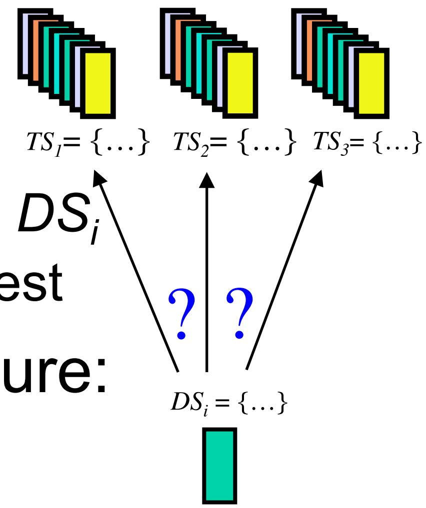
- **Corpus**
  - Training set WSJ 1987:
    - 16,137 texts (32 topics)
  - Test set WSJ 1988:
    - 12,906 texts (31 topics)
  - Texts indexed into categories by humans
- **Signature data**
  - 300 terms each, using *tf.idf*
  - Word forms: single words, demorphed words, multi-word phrases
- **Topic distinctness...**
  - Topic hierarchy

RANK	ARO	BNK	ENV	TEL
1	contract	bank	epa	at&t
2	air_force	thrift	waste	network
3	aircraft	banking	environmental	fcc
4	navy	loan	water	cbs
5	army	mr.	ozone	cable
6	space	deposit	state	bell
7	missile	board	incinerator	long-distance
8	equipment	fslic	agency	telephone
9	mcdonnell	fed	clean	telecomm.
10	northrop	institution	landfill	mci
11	nasa	federal	hazardous	mr.
12	pentagon	fdic	acid_rain	doctrine
13	defense	volcker	standard	service
14	receive	henkel	federal	news

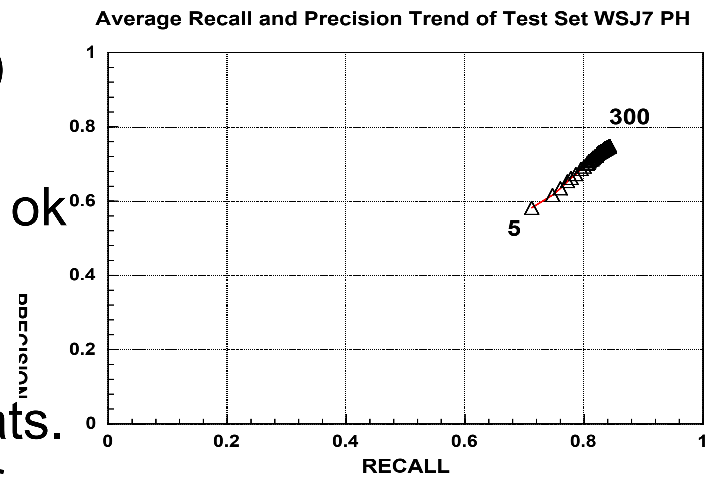


Evaluating signatures

- **Solution**: Perform text categorization task:
  - create N sets of texts, one per topic
  - create N topic signatures TS_k
  - for each new document, create document signature DS_i
  - compare DS_i against all TS_k ; assign document to best
- **Match function**: vector space similarity measure:
  - Cosine similarity, $\cos\theta = TS_k \cdot DS_i \ / \ |TS_k||DS_i|$



- **Test 1** (Hovy & Lin, 1997, 1999)
  - Training: 10 topics; ~3,000 texts (TREC)
  - Contrast set (background): ~3,000 texts
  - Conclusion: *tf.idf* and χ^2 signatures work ok but depend on signature length
- **Test 2** (Lin & Hovy, 2000):
  - 4 topics; 6,194 texts; uni/bi/trigram signats.
  - Evaluated using SUMMARIST: λ > *tf.idf*



Text pollution on the web

Goal: Create word families (signatures) for *each concept in the Ontology*. Get texts from Web

Main problem: text pollution. Which search term?

<MORTICE,w=33.7982>	<STAR, w=75.1358>	<AIRCRAFT, w=207.998>
<WOODWORKING, w=20.9227>	<ORION,w=55.8937>	<ENGINE, w=178.677>
<TENNON, w=20.9227>	<PYRAMID,w=42.1494>	<WING, w=138.36>
<JOINERY, w=17.7038>	<DNA,w=41.2331>	<PROPELLER, w=122.317>
<WOOD, w=15.8356>	<SOUL,w=31.1539>	<FLY, w=103.187>
<HARDWOOD, w=14.4849>	<IMPLOSION,w=23.8236>	<AIRPLANE, w=98.0431>
<JASON, w=14.4849>	<KHUFU,w=19.3133>	<AVIATION, w=96.5663>
<DOTH, w=12.8755>	<GOLD,w=18.3897>	<FLIGHT, w=85.3079>
<BRASH, w=12.8755>	<RECURSION,w=18.3258>	<AIR, w=80.1996>
<OAK, w=12.8281>	<BELLATRIX,w=17.7038>	<WARBIRDS, w=72.4247>
<WEDGE, w=11.9118>	<OSIRIS,w=17.7038>	<PILOT, w=71.4707>
<FURNITURE, w=10.0792>	<PHI,w=16.4932>	<MPH, w=65.987>
<TOOL, w=9.19486>	<EMBED,w=16.4932>	<CONTROL, w=65.9729>
<SHAFT, w=8.17321>	<MAGNETIC,w=16.4932>	<FUEL, w=62.3078>

Purifying: Later work: used Latent Semantic Analysis

Purifying with Latent Semantic Analysis

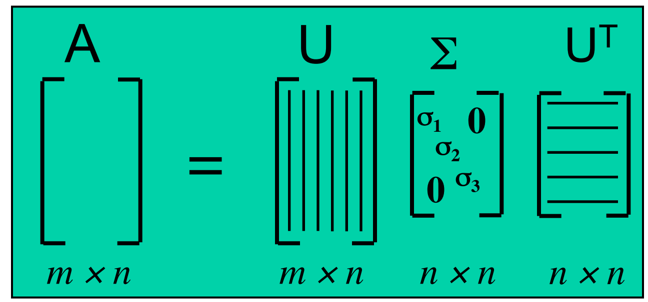
- Technique used in Psychologists to determine basic cognitive conceptual primitives (Deerwester et al., 1990; Landauer et al., 1998).
- Singular Value Decomposition (SVD) used for text categorization, lexical priming, language learning…
- LSA automatically creates collections of items that are correlated or anti-correlated, with strengths:

  ice cream, drowning, sandals ⇨ *summer*
- Each such collection is a 'semantic primitive' in terms of which objects in the world are understood.

- We tried LSA to find most reliable signatures in a collection— reduce number of signatures in contrast set.

LSA for signatures

- Create matrix A, one signature per column (words × topics).
- Apply SVDPAC to compute U so that $A = U \Sigma U^T$:
  - U : $m \times n$ orthonormal matrix of left singular vectors that span space
  - U^T : $n \times n$ orthonormal matrix of right singular vectors
  - Σ : diagonal matrix with exactly *rank(A)* nonzero singular values; $\sigma_1 > \sigma_2 > \ldots > \sigma_n$



- Use only the first *k* of the new concepts: $\Sigma' = \{\sigma_1, \sigma_2 \ldots \sigma_k\}$.
- Create matrix A′ out of these *k* vectors: $A' = U \Sigma' U^T \approx A$.

A′ is a new (words × topics) matrix, with different weights and new 'topics'. Each column is a purified signature.

Some results with LSA

(Hovy and Junk 99)

- Contrast set (for *idf* and χ^2): set of documents on very different topic, for good *idf*
- Partitions: collect documents within each topic set into partitions, for faster processing. */n* is a collecting parameter
- U function: function for creation of LSA matrix

Results:

- Demorphing helps
- χ^2 better than *tf* and *tf.idf*
- LSA improves results, but not dramatically

TREC texts

Function	Demorph?	Partitions	U function	Recall	Precision
Without contrast set					
tf	no			0.748447	0.628782
tf	yes			0.766428	0.737976
tf	yes	10	*tf*	0.820609	0.880663
tf	yes	20	*tf*	0.824180	0.882533
tf	yes	30	*tf*	0.827752	0.884352
With contrast set					
tf.idf	no	10	*tf.idf*	0.626888	0.681446
tf.idf	no	20	*tf.idf*	0.635875	0.682134
tf.idf	yes	10	*tf.idf*	0.718177	0.760925
tf.idf	yes	20	*tf.idf*	0.715399	0.762961
X^2	no	10	X^2	0.847393	0.841513
X^2	no	20	X^2	0.853436	0.849575
X^2	yes	10	X^2	0.822615	0.828412
X^2	yes	20	X^2	0.839114	0.839055
Varying partitions					
X^2	yes	30/0	X^2	0.912525	0.881494
X^2	yes	30/3	X^2	0.903534	0.879115
X^2	yes	30/6	X^2	0.903611	0.873444
X^2	yes	30/9	X^2	0.899407	0.868053

Web signature experiment

Procedure:

1. Create query from Ontology concept (word + defn. words)
2. Retrieve ~5,000 documents (8 web search engines)
3. Purify results (remove duplicates, html, etc.)
4. Extract word family (using *tf.idf*, χ^2, LSA, etc.)
5. Purify
6. Compare to siblings and parents in the Ontology

Problem: raw signatures overlap…

- average parent-child node overlap: ~50%
- Bakery—Edifice: ~35% …too far: missing generalization.
- Airplane—Aircraft: ~80% …too close?

Remaining problem: web signatures still not pure...

WordNet: In 2002–04, Agirre and students (U of the Basque Country) built signatures for all WordNet nouns

Later work using signatures

- **Multi-document summarization** (Lin and Hovy, 2002)
  - Create λ signature for each set of texts
  - Create IR query from signature terms; use IR to extract sentences
  - (Then filter and reorder sentences into single summary.
  - **Performance:** DUC-01: tied first; DUC-02: tied second place
- **Wordsense disambiguation** (Agirre, Ansa, Martinez, Hovy 2001)
  - Try to use WordNet concepts to collect text sets for signature creation: *(word+synonym > def-words > word .AND. synonym .NEAR. def-word >* etc…)
  - Built competing signatures for various noun senses: (a) WordNet synonyms; (b) SemCor tagged corpus (χ^2); (c) web texts (χ^2); (d) WSJ texts (χ^2)
  - **Performance**: Web signatures > random, WordNet baseline.
- **Email clustering** (Murray and Hovy)
  - Social Network Analysis: Cluster emails and create signatures
  - Infer personal expertise, project structure, experts omitted, etc.
  - Corpora: ENRON (240K emails), ISI corpus, NSF eRulemaking corpus

Part 3

LEARNING INSTANCES

Collaborators

- Zornitsa Kozareva, grad student at U of Alicante, during a visit to ISI 2007–08; joined ISI in August 2009
- (Ellen Riloff, U of Utah, on sabbatical at ISI in 2007–08)
- Eduard Hovy, ISI

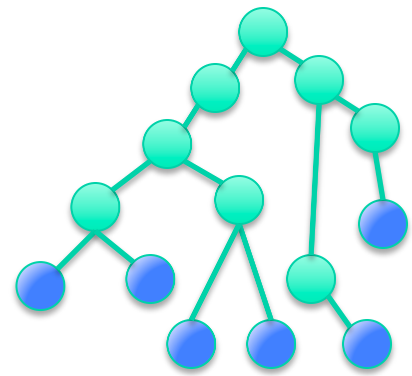
Question

Using text on the web,
can you automatically build a domain-specific ontology, plus its instances, on demand?

- Instance data
- Metadata (type hierarchies)
- Relation values (attribute data)

The challenge

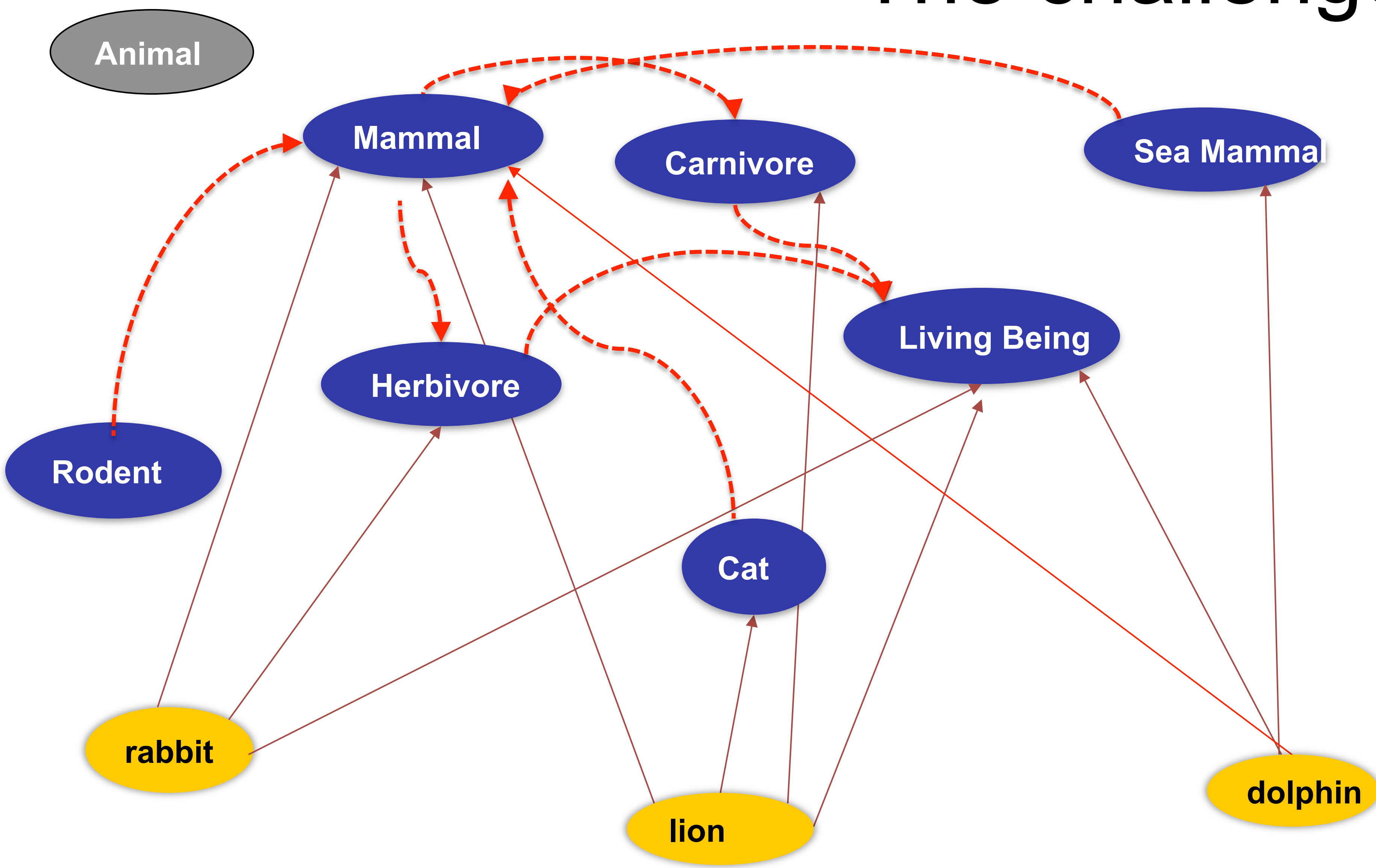
- For a given domain, can we learn its structure (metadata) and instances simultaneously?



- That is, can we learn…
  - instance/basic level terms?
  - non-instance terms and organization?

…with no (or minimal) supervision, using automatic knowledge acquisition methods, all together (so one type helps the other)?

The challenge



Some problems

- Some things are hard to get right: determine correctness (Precision)
- Some things are hard to encompass: determine coverage (Recall)
- Some things are hard to organize: determine reasonable schema (metadata/ taxonomy)
- People lie: Determine data trustworthiness
- Things change: Determine recency / timeliness

Related ontology-related work

- Based on the knowledge extracted
  - Hypernyms and other relations (Hearst 92; Ravichandran and Hovy 02; Paşca 04; Etzioni et al. 05; Kozareva et al. 08; Ritter et al. 09)
  - Instances (Paşca and Van Durme 08)
- Based on the techniques employed
  - Lexico-syntactic patterns (Riloff and Jones 99; Fleischman and Hovy 02)
  - Unsupervised clustering (Lin 98; Lin and Pantel 02; Davidov and Rapoport 06; Suchanek et al. 07, Snow and Jurafsky 08)
- Automatic ontology construction (Caraballo 99; Cimiano and Volker 05; Mann 05)

Approach and definitions

- Start with instances / basic level terms
- Then learn non-instance / organizational terms
- Then taxonomize, in stages
- Then learn inter-concept relations

- **Term**: English word
- **Concept**: Any item in classification taxonomy
- **Class**: Concept in taxonomy, but above Basic Level
- **Basic level concept**: Concept at Basic Level in Prototype Theory (Rosch 78): *dog* (not *mammal* or *collie*); *car* (not *vehicle* or '*BMW 520i*')
- **Instance**: More precise then concept: single individual entity (*Lassie*, *Aslan*; '*BMW 520i with reg EX740N*')

Hyponym pattern mining

- Inspired by Hearst,1992 hyponym patterns (Pasca04; Etzioni et al.,05; Pasca07)

  “ *class_name* such as * ”
- Sentences contain clues as to their meanings

  countries such as **France** have regulated economic life
- Combination of lexico-syntactic information or statistical evidence, but still the quality of acquired information is insufficient

Overall plan

- **Goal**: Develop (semi-)automated ways of building (small) term taxonomies from domain texts / the web
- **Three-step approach**:
  1. Collect related terms
  2. Organize them into small taxonomies
  3. Add features

- Related work:
  - Initial work (Hearst 1992): NP patterns signal hyponymy:

    "NP_0 such as NP_1, NP_2…"

    "NP_0, especially NP_1…"

    "NP_0, including NP_1, NP_2, etc."
  - Much subsequent work using different patterns for different relations — part-whole (Girju et al. 2006), named entities (Fleischman and Hovy 2002; Etzioni et al., 2005), other relations (Pennacchiotti and Pantel, 2006; Snow et al, 2006; Paşca and Van Durme, 2008), etc.
- Main problem: classes are small, incomplete, and noisy

Step 1: Instances (Kozareva et al., ACL 08)

- Define doubly-anchored pattern (DAP); extends (Hearst 92) hyponym pattern:

  [NP_0 such as NP_1 and ?]

- Collect terms:

  animals such as lions and *

  using algorithm '*reckless bootstrapping*':

  Start with seed term NP_0 and one instance (or Basic Level concept) NP_1,

  learn more terms in position *: NP_2, NP_3, …

  Then, replace NP_1 by NP_2, NP_3 ,… , and learn more NP_i

  … repeat

Doubly-anchored pattern (DAP)

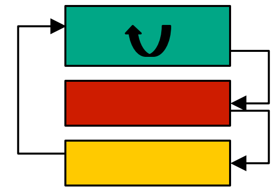
(Kozareva et al., ACL 2008)

- Doubly-anchored pattern, extending Hearst's hyponym pattern:

  [***class_name*** such as ***class_member*** and *****]

  - *class_name* is the name of the semantic class to be learned
  - *class_member* is a (given) example of the semantic class
  - (*) indicates the location of the extracted terms

Knowledge Harvesting Algorithm



0. Start with instance / basic level term
1. Learn more instances / basic level concepts
   – Use DAP pattern in bootstrapping loop:

*animals such as lions and ** → *tigers*, *bears*, *unicorns*, *…*

2. Learn non-instance terms (classes)
   – Use DAP^{-1} pattern with learned instances:

beasts, *stuffed toys*, *mammals*, *…* ← ** such as lions and tigers*

3. Position learned concepts using DAP pattern

*freq(A such as B and *) > freq(B such as A and *) => B isa A*

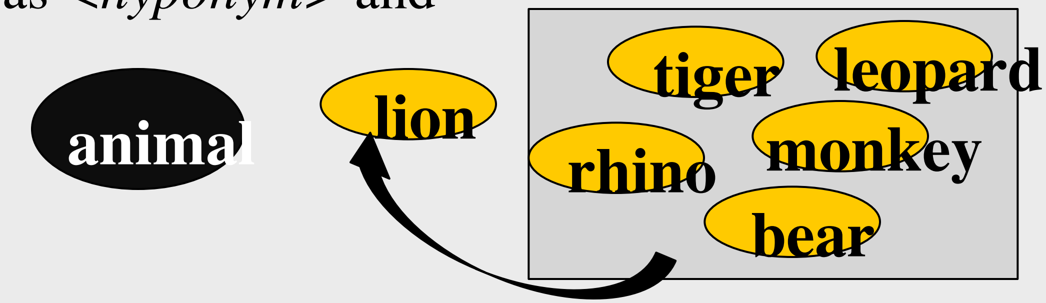
< one RootCategory , one Seed Instance > **Step 1**

Instance Harvesting

- DAP pattern:

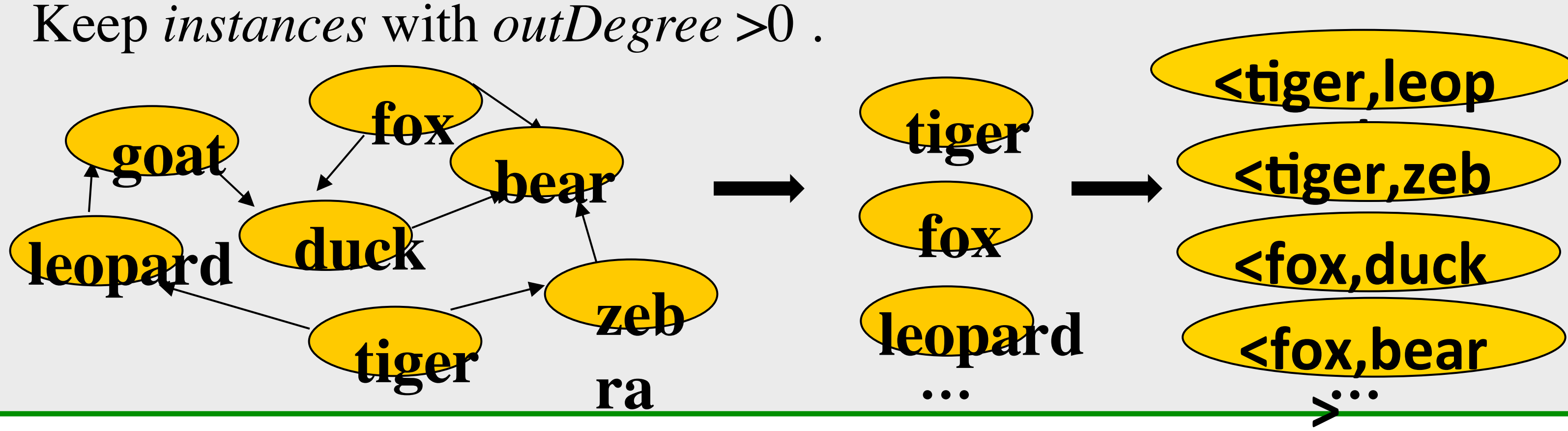
  <*hypernym*> such as <*hyponym*> and *

- Breadth-first search.



Instance Ranking

- Build directed Hyponym Pattern Linkage Graph of *instances*.
- Rank *instances* by *outDegree*, where *outDegree(v)* of a node *v* is the sum of all outgoing edges from *v* normalized by *V-1*.
- Keep *instances* with *outDegree* >0 .

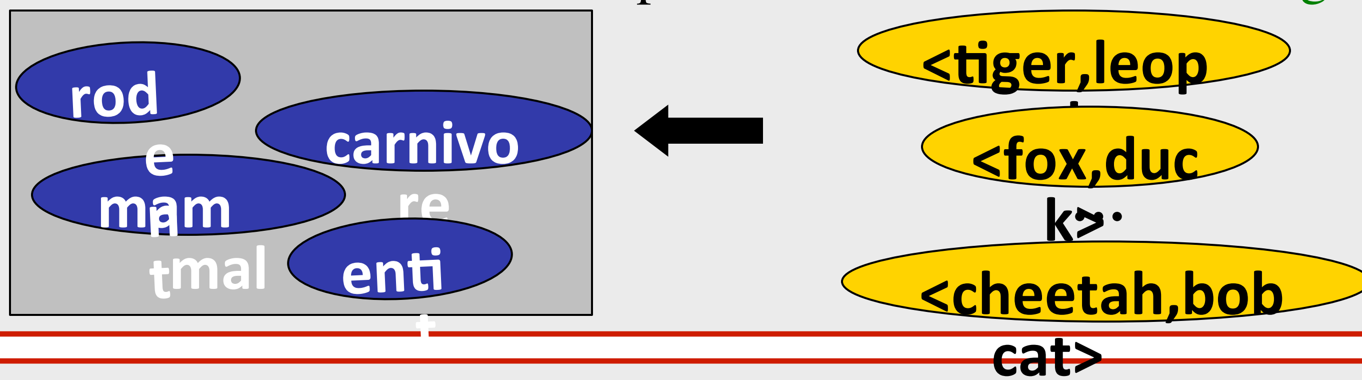


Intermediate Concept Harvesting

- DAP^{-1} pattern:

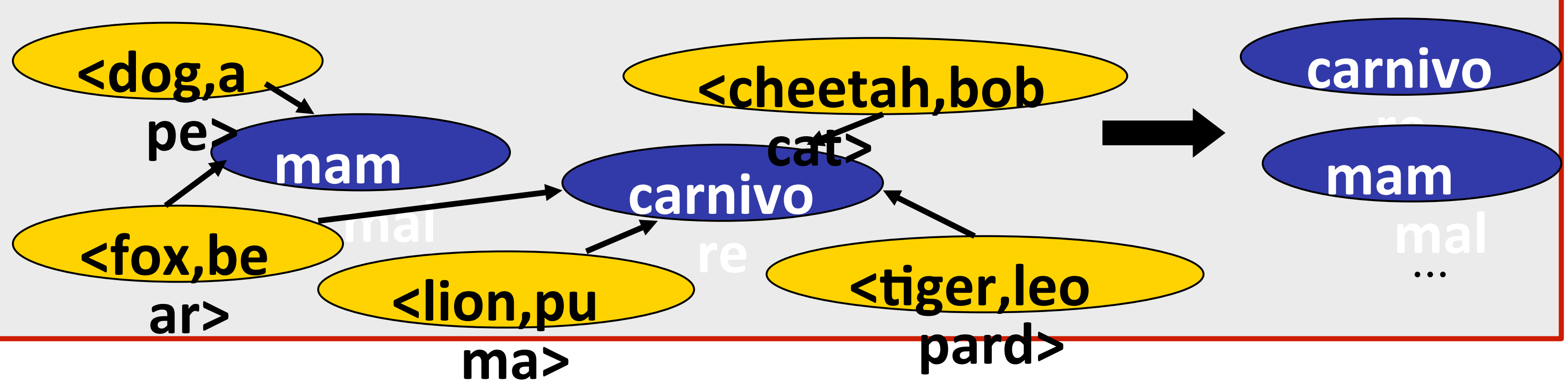
  * such as <*hyponym1*> and <*hyponym2*>

- Exhaustive search of all instance pairs from Instance Harvesting.

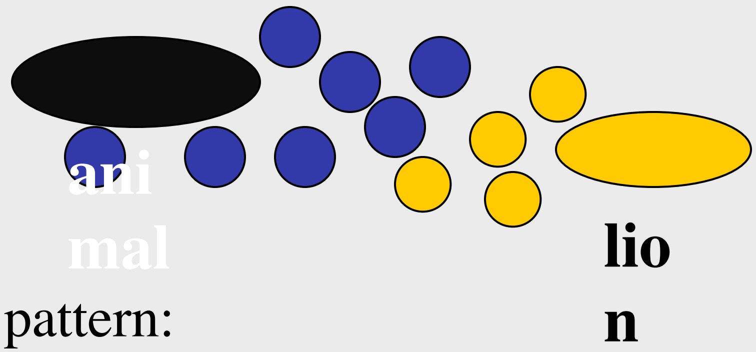


Intermediate Concept Ranking

- Build graph of *concepts* and <*instance,instance*> pairs.
- Rank *concepts* by *inDegree*, where *inDegree(c)* of a *concept* c is the sum of all incoming edges of the instance pairs normalized by *V-1*.



Concept Positioning Test

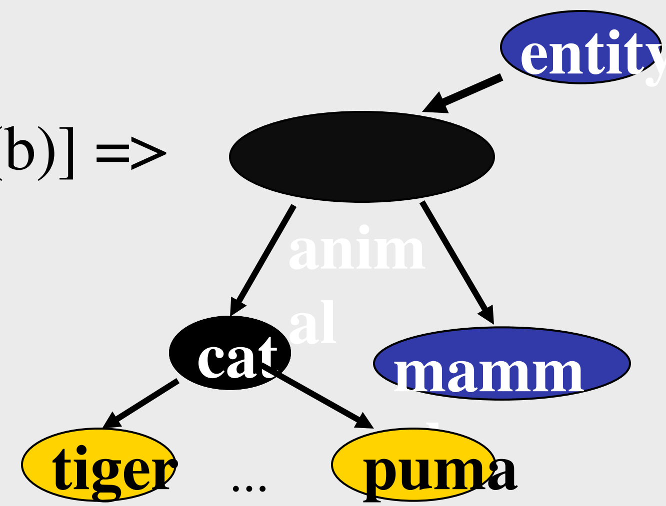


- DAP pattern:

  freq(a) = <*conceptA*> such as <*conceptB*> and *

  freq(b) = <*conceptB*> such as <*conceptA*> and *

if [freq(a) > freq(b)] =>



In addition to **animals such as lions and** elephants, Aesop composed stories using animals in the environment of ancient Greece such as the fox, the owl, ...
www.shvoong.com/books/71810-aesop-fables/ - 57k - Cached - Similar pages

Some of the monsters that medieval artists drew along the edges of manuscript pages were based on real **animals, such as lions and** whales. ...
www.carmenbutcher.com/carmenbutcher/Handouts/

Jun 25, 2008 ... are several Cald creatures that are animated bodies of armor, and several are based on jungle **animals, such as lions and** alligators. ...
en.wikipedia.org/wiki/Magi_Nation_(GBC) - 29k - Cached - Similar pages

... horses, cows, llamas, sheep, and pigs; wild animals, such as eagles, hawks, and squirrels; and occasional zoo **animals, such as lions and** kangaroos. ...
www.countrylines.com/2006/10/05/the-vet-is-in/ - 21k - Cached - Similar pages

Skeptics paint a picture of Noah going to countries remote from the Middle East to gather **animals such as kangaroos and** koalas from Australia, ...
www.christiananswers.net/q-aig/aig-c006.html - 26k - Cached - Similar pages

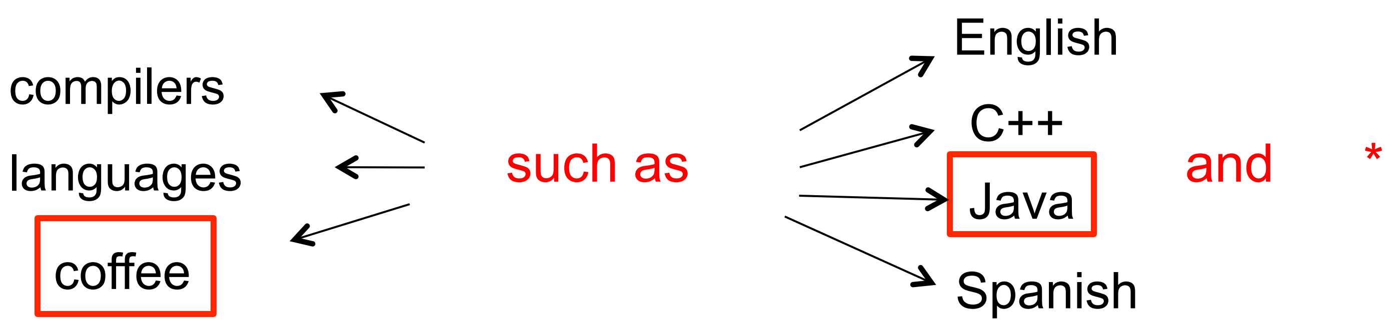
Hand feeding of the free-ranging **animals such as kangaroos and** water birds. Meet some rare and endangered North Queensland wildlife, including cassowaries, ...
www.billabongsanctuary.com.au/booknow.html - 22k - Cached - Similar pages

today we got up and close with the (tame) wildlife. we went to a sanctuary where we could walk among the cute **animals, such as kangaroos and** peacocks and ...
monsuuni.blogspot.com/2006/07/getting-to-know-locals-animals.html - 16k -

Experiment mixing some **animals, such as peacocks and** tigers, together in an enclosure, it might prove to be quite interesting! But the game expects you to ...
www.theurbanwire.com/jan05/zootycoon2print.html - 9k - Cached - Similar pages

Power of DAP

- Virtually eliminates ambiguity, because *class_name* and *class_member* mutually disambiguate each other

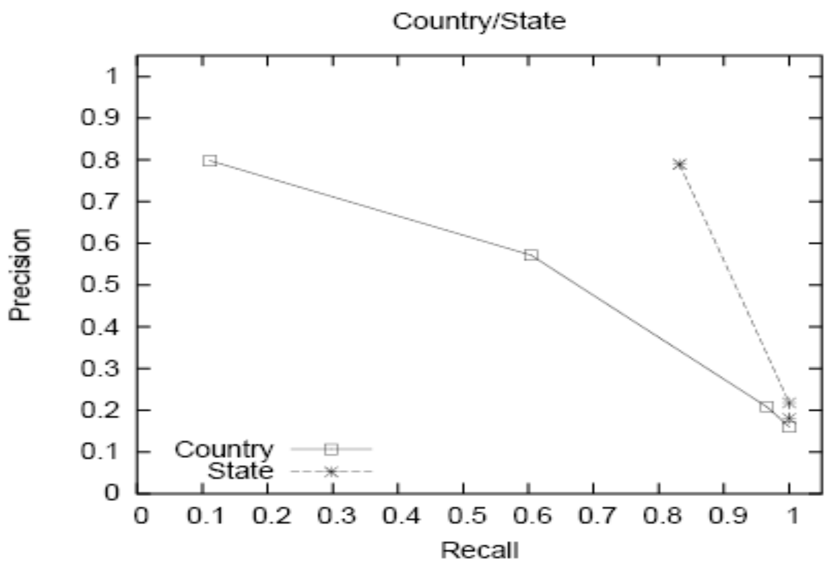


- So, more likely to generate results of desired type

- Not perfect, though:

A garden without botanist, a project that provides for the introduction of **animals such as peacocks, and** the master plan, defended the president of the ...
www.accommodation.io/index.php?view_page=news&acticle=294&lang=1&changelang=1 - 58k

Performance of reckless bootstrapping



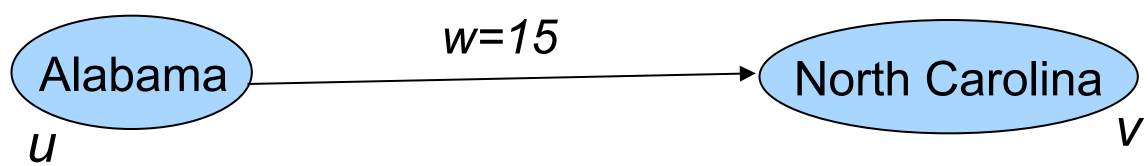
Iter.	countries	states	singers	fish
1	.80	.79	.91	.76
2	.57	.21	.87	.64
3	.21	.18	.86	.54
4	.16	-	.83	.54

Problem: search needs guidance
Solution: rank learned instances

Hyponym pattern linkage graphs

- HPLG=(V,E) where vertex $v \in V$ is an instance, and $e \in E$ is an edge between two instances

*Some **states, such as** __**Alabama**__ **and** __North Carolina__, provide…*



- Weight *w* of edge is frequency with which *u* generates *v*
- Growing the graph:
– Compute score for each vertex $\{u_{2i}\}$
– Try various scoring formulas
– On each iteration, take for next v_1 only *highest-scoring unexplored node* from $\{u_{2i}\}$

Guiding the growth: Scoring

- Apply measures separately or combined
  - **Popularity**: Ability of term to be discovered by other terms
    - in-Degree (inD) of a node (v) is the sum of the weights of all incoming edges (u,v), where (u) is a trusted member, normalized by V-1
    - Best edge (BE) of a node (v) is the maximum edge weight among the incoming edges (u,v), where u is a trusted member
    - Key Player Problem (KPP) high KPP indicates strong connectivity and proximity to the rest of the nodes

$$KPP(v) = \frac{\sum_{u \in V} \frac{1}{d(u,v)}}{|V| - 1}$$

  - **Productivity**: Ability of term to discover other terms
    - outDegree (outD) of a node (v) is the sum of all outgoing edges from v normalized by V-1
    - totalDegree (totD) of a node (v) is the sum of inDegree and outDdegree edges of v normalized by V-1
    - betweenness (BE), where σ_{st} is the number of shortest paths from s to t, and $\sigma_{st}(v)$ is the number of shortest paths from s to t that pass through v
    - PageRank (PR)

$$BE(v) = \sum_{\substack{s \neq v \neq t \in V \\ s \neq t}} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

$$PR(v) = \frac{(1-\alpha)}{|V|} + \alpha \sum_{u,v \in E} \frac{PR(u)}{\mathrm{outD(u)}}$$

Test examples of learning

- Explore the learning power of HPLG with different size classes
  - closed: **countries** (194 elements), **USA states** (50 elements)
  - open: **fishes** (gold standard Wikipedia), **singers** (manually reviewed)
- Validate performance of each class independently with five randomly selected seeds; then measure average performance

Performance: Closed-class

number of learned instances

States			
	Popularity		
N	BE	KPP	inD
25	1.0	1.0	1.0
50	.96	.98	.98
64	.77	.78	.77

dynamic graph

BE – best edge
KPP – key player problem
inD – in-Degree

Performance: Closed-class

States						
	Popularity			Pop&Prd		
N	BE	KPP	inD	totD	BT	PR
25	1.0	1.0	1.0	**1.0**	.88	.88
50	.96	.98	.98	**1.0**	.86	.82
64	.77	.78	.77	.78	.77	.67

precompiled graph

BE – best edge
KPP – key player problem
inD – in-Degree
totD – total degree
BT – betweenness
PR – Page Rank

Performance: Closed-class

States							
	Popularity			Pop&Prd			Prd
N	BE	KPP	inD	totD	BT	PR	outD
25	1.0	1.0	1.0	**1.0**	.88	.88	**1.0**
50	.96	.98	.98	**1.0**	.86	.82	**1.0**
64	.77	.78	.77	.78	.77	.67	.78

BE – best edge
KPP – key player problem
inD – in-Degree
totD – total degree
BT – betweenness
PR – Page Rank

- HPLGs perform better than reckless bootstrapping
- *outD* and *totD* discover all state members
- BUT if there are only 50 USA states, why does the algorithm keep on learning?

The extra 14 states…

- The ‘leakage’ effect:
  - “…Southern states such as **Florida and <u>Georgia</u>** are…”
  - “…former Soviet states such as **<u>Georgia</u> and Ukraine** always…”

…which leads to:
  - **<u>Georgia</u>, Ukraine**, Russia, Uzbekistan, Azerbaijan, Moldava, Tajikistan, Armenia, Chicago, Boston, Atlanta, Detroit, Philadelphia, Tampa, Moldavia …

- Here, due to ambiguity of “Georgia”. But not always…
  - “Findlay now has over 20 restaurants in states such as **Florida and Chicago**”

Performance: Open-class

Fish		
	Pop	Prd
N	KPP	outD
10	.90	**1.0**
25	.88	**1.0**
50	.80	**1.0**
75	.69	.93
100	.68	.84
116	.65	.80

Performance: Open-class

Fish		
	Pop	Prd
N	KPP	outD
10	.90	**1.0**
25	.88	**1.0**
50	.80	**1.0**
75	.69	.93
100	.68	.84
116	.65	.80

Singers		
	Pop	Prd
N	inD	outD
10	.92	**1.0**
25	.91	**1.0**
50	.92	.97
75	.91	.96
100	.89	.96
150	.88	.95
180	.87	.91

Performance: Open-class

Fish		
	Pop	Prd
N	KPP	outD
10	.90	**1.0**
25	.88	**1.0**
50	.80	**1.0**
75	.69	.93
100	.68	.84
116	.65	.80

Singers		
	Pop	Prd
N	inD	outD
10	.92	**1.0**
25	.91	**1.0**
50	.92	.97
75	.91	.96
100	.89	.96
150	.88	.95
180	.87	.91

Countries		
	Pop	Prd
N	inD	outD
50	.98	**1.0**
100	.94	**1.0**
150	.91	**1.0**
200	.83	.90
300	.61	.61
323	.57	.57

Error analysis

- type 1: incorrect proper name extraction
- type 2: instances that formerly belonged to the semantic class
- type 3: spelling variants
- type 4: sentences with wrong factual assertions
- type 5: broken expressions

Comparison with recent work

- (Paşca et al., 2007)

generated instances (country)	Pasca 07 (precision)	DAP outDegree (precision)
100	95%	100%
150	82%	100%

- KnowItAll (Etzioni et al., 2005)

country	KnowItAll 1	KnowItAll 2	DAP outDegree
Prec.	79%	97%	100%
Rec.	89%	58%	77%

Part 4

LEARNING CLASSES

Step 2: Classes

(Hovy et al. EMNLP 09)

- Now DAP^{-1}: use DAP in ‘**backward**’ direction:

[? such as NP$_1$ and NP$_2$]

e.g.,

*** such as lions and { tigers | peacocks | … }**

*** such as peacocks and { lions| snails | … }**

using algorithm:

1. Start with terms NP$_1$ and NP$_2$, learn more classes at *
2. Replace NP$_1$ and/or NP$_2$ by NP$_3$,… , and learn additional classes at *

… repeat

He trained and performed with animals, **such as lions and tigers**, for many years. He trained animals to come out when they hear a music tape. ...
en.allexperts.com/q/Wild-Animals-705/Tiger-lion-cubs.htm - 22k - Cached - Similar pages

She said a 2004 Oxford University study concluded that long- ranging carnivores, **such as lions and tigers**, suffer in captivity and show signs of serious ...
findarticles.com/p/articles/mi_qn4176/is_20071228/ai_n21180527 - 44k -

... Centre for Cellular and Molecular Biology (CCMB) will adopt cloning techniques to preserve endangered species **such as lions and tigers** in the country. ...
www.indianexpress.com/res/web/ple/ie/daily/19990104/0045075.html - 19k -

Some predators, **such as lions and tigers**, are large and ferocious, while others can be small and benign in appearance, such as lady bugs. ...
www.biologyreference.com/Po-Re/Predation-and-Defense.html - 18k - Cached - Similar pages

May 10, 2007 ... Canids such as wolves, jackals, coyotes and foxes (except domestic dogs); Felids **such as lions and tigers** (except domestic cats); Bears ...
www.vancouverhumanesociety.bc.ca/issues/exotic_animals/general - 26k -

fur-bearing animals—is not required to. inspect or regulate products made from. domesticated animals **such as dogs and cats**. ...
files.hsus.org/web-files/PDF/betrayal_trust.pdf - Similar pages

They are also some of the most familiar organisms, including pets **such as dogs and cats**, as well as many farm and work animals, such as sheep, cattle, ...
bubl.ac.uk/link/c/cats.htm - 11k - Cached - Similar pages

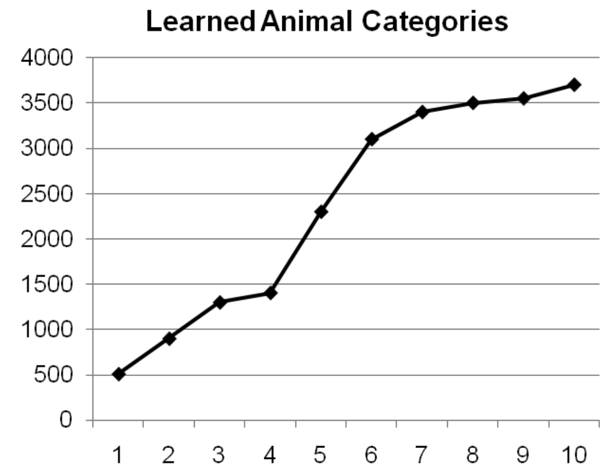
that are not mentioned specifically in the ordinance, **such as dogs and cats**, the identification. 'hair' suffices without the species being cited. ...
www.infurmation.com/pdf/dutchparl03ae.pdf - Similar pages

Experiment 1: Interleave DAP and DAP^{-1}

- Seeds: *Animals—lions* and *People—Madonna* (seed term determines Basic Level or instance)
- Procedure:
  - Sent DAP and DAP^{-1} queries to Google
  - Collected 1000 snippets per query, kept only unique answers (counting freqs)

    (for DAP^{-1}, extracted 2 words in target position)
  - Algorithm ran for 10 iterations
- Results: 1.1 GB of snippets for Animals and 1.5 GB for People:
  - 913 Animal basic-level concepts and 1,344 People instances with *Out-Degree > 0*

Results 1



- Found staggering variety of terms:
  - Growth doesn't stop!
  - Example animals:

    accessories, activities, agents, amphibians, animal_groups, animal_life, amphibians, apes, arachnids, area, …, felines, fish, fishes, food, fowl, game, game_animals, grazers, grazing_animals, grazing_mammals, herbivores, herd_animals, household_pests, household_pets, house_pets, humans, hunters, insectivores, insects, invertebrates, laboratory_animals, …, water_animals, wetlands, zoo_animals
- Much more diverse than expected:
  - Probably useful: *laboratory animals, forest dwellers, endangered species …*
  - Useful?: *bait, allergens, seafood, vectors, protein, pests …*
  - What to do?: *native animals, large mammals …*
- Problem: How to evaluate this?

Evaluation: Are the learned classes really Animals / People?

- Examples (top 10):

#Ex.	AnMem	AnCat	PMem	PCat
1	dogs	insect	Jesse Jackson	leader
2	kudu	bird	Paris Hilton	reformer
3	cats	specie	Bill Clinton	celebrity
4	sheep	invertebrate	Bill Gates	prophet
5	rats	predator	Brad Pitt	artist
6	mice	mammal	Moses	star
7	rabbits	pest	Tiger Woods	dictator
8	horses	pet	Gandhi	writer
9	pigs	crustacean	Donald Trump	teacher
10	cows	herbivore	Picasso	poet

- Subclasses/instances:
  - **Animals** (evaluate against lists compiled from websites):

Iteration	1	2	3	4	5	6	7	8	9	10
Accuracy	0.79	0.79	0.78	0.70	0.68	0.68	0.67	0.67	0.68	0.71

  - **People** (ask human judges):

	Judge1	Judge2	Judge3
Person	190	192	189
NotPerson	10	8	11
Accuracy	0.95	0.96	0.95

New classes generate new instances

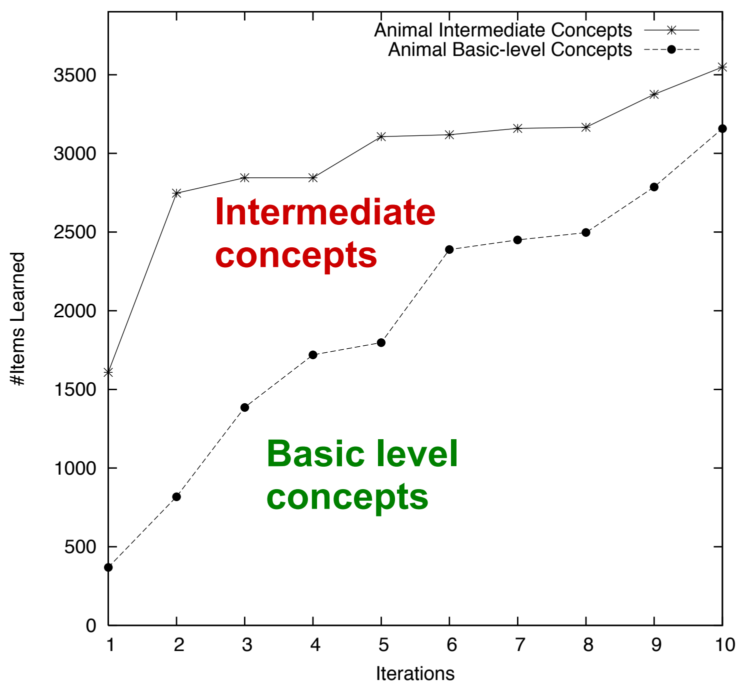
- New classes from DAP^{-1} provide additional seed terms for DAP

  …now can reach instances and basic level concepts not found by DAP alone:

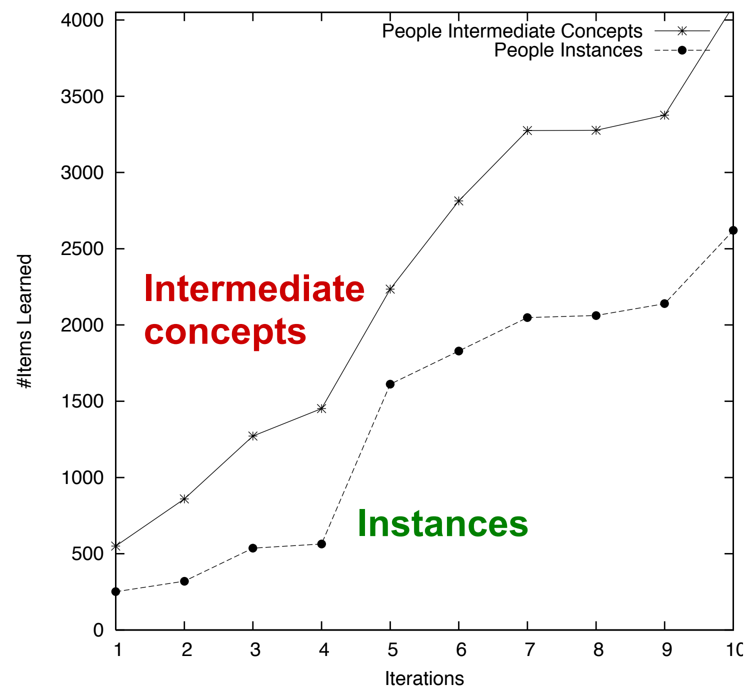
  - "animals such as lions and *" ➔ lion-like animals
  - "herbivores such as antelope and *" ➔ kudu, etc.

Results 2

Surprisingly, found many more classes than instances:



Animals



People

Evaluation woes: Precision

- Would like to evaluate against WordNet or Wikipedia (international standards, available, large, etc.)
- BUT:
  - They do not contain many of our learned terms (even though many are sensible and potentially valuable)
  - Point of our work is to learn more/new concepts than currently available

- Other projects create ad hoc measures:
  - E.g.: Ritter et al. learn that { *jaguar* is-a: *animal, mammal, toy, sports-team, car-make, operating-system* } and count <u>all</u> correct — even if not *Animal*

- Our strategy:
  - Count only correct classes
  - Compare against WordNet **<u>and</u>** do manual evaluation (if possible)

Evaluation woes: Recall

- Cannot easily compare to WordNet:
  - Doesn't indicate Basic Level
  - Doesn't include Instances (very few proper names)
- So, need to ask people … this is expensive

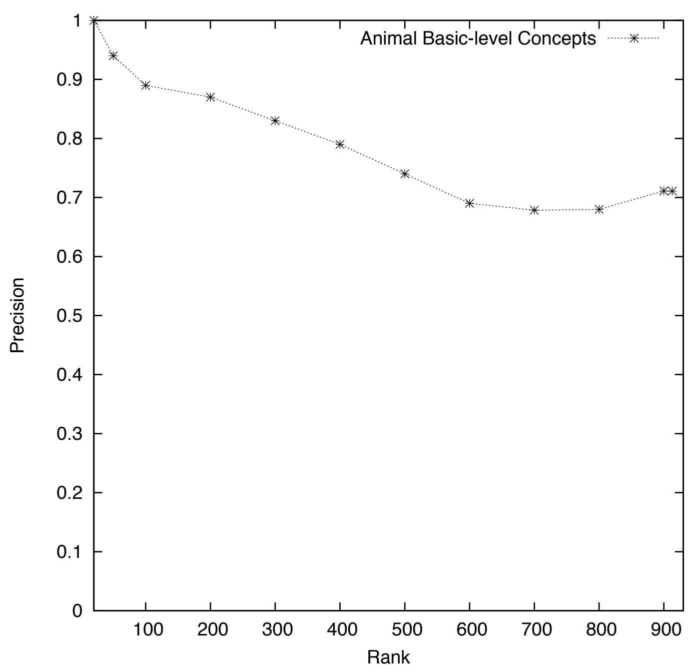
Evaluation measures

- **Precision**:
  - $Pr_{WN} = \dfrac{\text{\#terms found in WordNet}}{\text{\#terms harvested by system}}$
  - $Pr_{HUM} = \dfrac{\text{\#terms judged correct by human}}{\text{\#terms harvested by system}}$

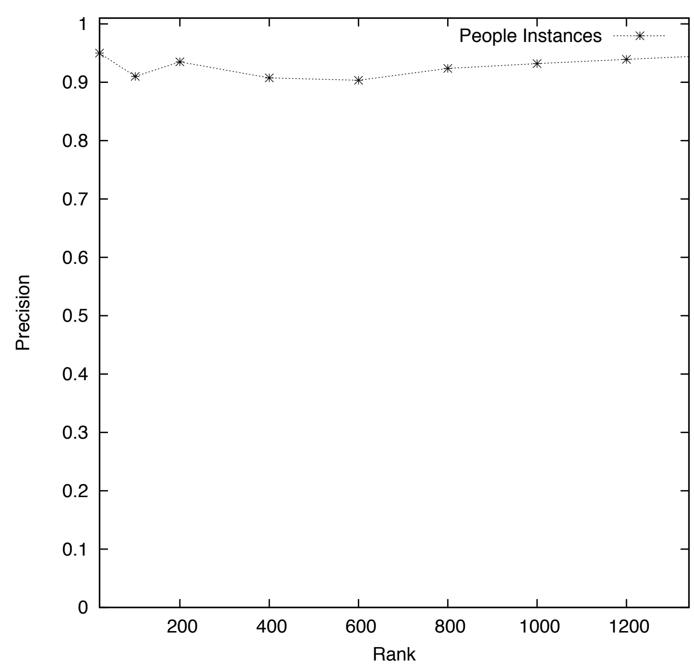
- **Recall substitute**:
  - NotInWN = #terms judged correct by human but not in WordNet

Evaluation #1: Basic terms and Instances

	# harvested	Pr_{WN}	Pr_{HUM}	NotInWN
Animals	913	.79	.71	48
People	1344	.23	.95	986



Animals: Precision at Rank N



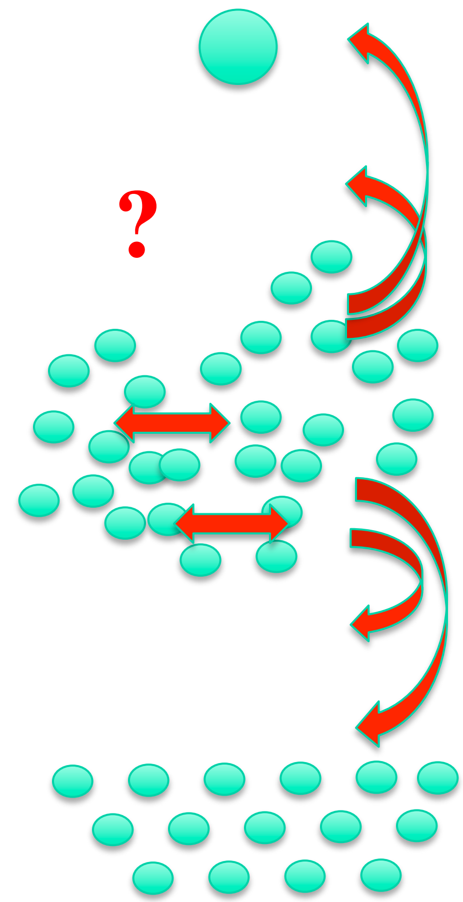
People: Precision at Rank N

Part 5

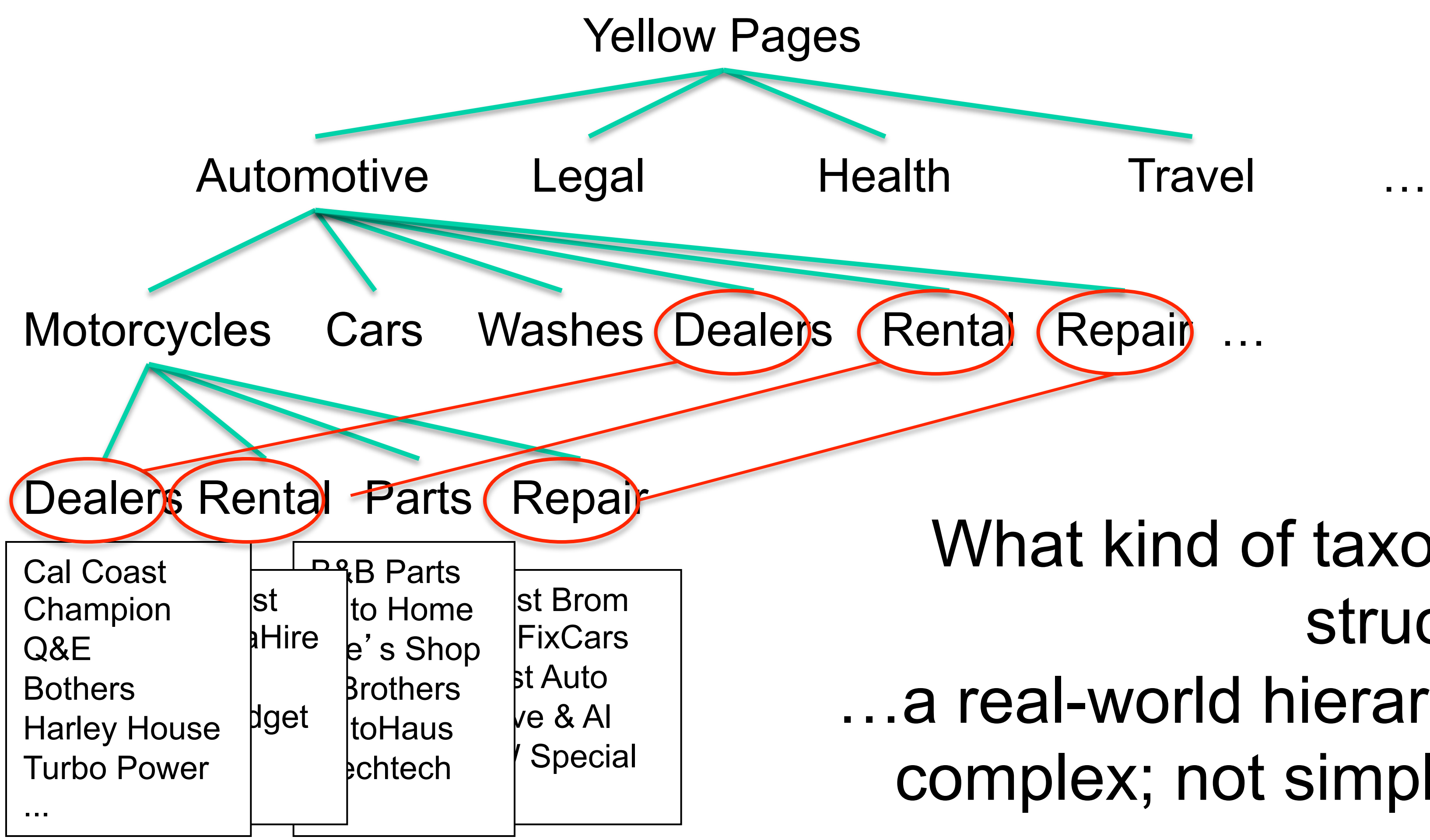
LEARNING TAXONOMY STRUCTURE

Challenge: Taxonomizing classes

- **Start**: animals
- **NP$_0$**: *amphibians apes … felines fish fishes food fowl game game_animals grazers grazing_animals grazing_mammals herbivores herd_animals household_pests household_pets house_pets humans hunters insectivores insects invertebrates laboratory_animals … monogastrics non-ruminants pets pollinators poultry predators prey … vertebrates water_animals wetlands zoo_animals*
- **NP$_2$**: *… alligators ants bears bees camels cats cheetahs chickens crocodiles dachshunds dogs eagles lions llamas … peacocks rats snails snakes spaniels sparrows spiders tigers turkeys varmints wasps wolves worms …*



Yahoo example



What kind of taxonomy structure?

…a real-world hierarchy is complex; not simple *is-a*

Experiment 2

- Re-ran algorithms in tandem (10 iterations)
  - Now learned 3,549 Animal and 4,094 People intermediate concepts
  - Filter: In-degree ranking and freq cutoff
- Evaluation:
  - Random sample of 437 Animal and 296 People concepts
  - Of these, 187 Animal concepts and 139 People concepts passed is-a (*Concept Positioning)* Test

Evaluating concepts

- First checked whether learned intermediate concepts are correct
  - Manually created small taxonomy to begin to group terms
  - Also included categories for wrong and dubious terms
- Then checked for ISA taxonomization using CPT

ANIMALS		
TYPE	**LABEL**	**EXAMPLES**
Correct	GeneticAnimal	*reptile,mammal*
	BehavioralByFeeding	*predator, grazer*
	BehaviorByHabitat	*saltwater mammal*
	BehaviorSocialIndiv	*herding animal*
	BehaviorSocialGroup	*herd, pack*
	MorphologicalType	*cloven-hoofed animal*
	RoleOrFunction	*pet, parasite*
Borderline	NonRealAnimal	*dragon*
	EvaluativeTerm	*varmint, fox*
	OtherAnimal	*critter, fossil*
BasicConcept	BasicAnimal	*dog, hummingbird*
NotConcept	GeneralTerm	*model, catalyst*
	NotAnimal	*topic, favorite*
	GarbageTerm	*brates, mals*

PEOPLE		
TYPE	**LABEL**	**EXAMPLES**
Correct	GeneticPerson	*Caucasian, Saxon*
	NonTransientEventRole	*stutterer, gourmand*
	TransientEventRole	*passenger, visitor*
	PersonState	*dwarf, schizophrenic*
	FamilyRelation	*aunt, mother*
	SocialRole	*fugitive, hero*
	NationOrTribe	*Bulgarian, Zulu*
	ReligiousAffiliation	*Catholic, atheist*
Borderline	NonRealPerson	*biblical figure*
	OtherPerson	*colleagues, couples*
BasicConcept	BasicPerson	*child, woman*
	RealPerson	*Barack Obama*
NotConcept	GeneralTerm	*image, figure*
	NotPerson	*books, event*

ISA relationship tests

- **Concept Positioning Test:**
  (apply DAP twice, inverting terms)
  Count freqs of terms generated by each term pair

[animals such as lions and *] ?
[lions such as animals and *] ?

- **Concept Children Test:**
  – Count intersections of terms generated by each term pair

	animals	amphibians	animal	apes	arachnids	arthropods	beasts	birds	bovines	bugs	burden	camelids
animals(94)	(155|155)=155	(155|6)=6	(155|21)=21	(155|2)=2	(155|2)=2	(155|6)=6	(155|20)=19	(155|14)=14	(155|1)=1	(155|3)=3	(155|4)=4	(155|1)=1
	animals	amphibians	animal	apes	arachnids	arthropods	beasts	birds	bovines	bugs	burden	camelids
prey(88)	(44|155)=44	(44|6)=4	(44|21)=13	(44|2)=0	(44|2)=2	(44|6)=5	(44|20)=7	(44|14)=6	(44|1)=1	(44|3)=3	(44|4)=1	(44|1)=0
	animals	amphibians	animal	apes	arachnids	arthropods	beasts	birds	bovines	bugs	burden	camelids
mammals(83)	(67|155)=67	(67|6)=2	(67|21)=20	(67|2)=1	(67|2)=0	(67|6)=0	(67|20)=14	(67|14)=4	(67|1)=1	(67|3)=0	(67|4)=3	(67|1)=0
	animals	amphibians	animal	apes	arachnids	arthropods	beasts	birds	bovines	bugs	burden	camelids
predators(83)	(53|155)=53	(53|6)=4	(53|21)=11	(53|2)=0	(53|2)=1	(53|6)=3	(53|20)=10	(53|14)=7	(53|1)=0	(53|3)=2	(53|4)=0	(53|1)=0
	animals	amphibians	animal	apes	arachnids	arthropods	beasts	birds	bovines	bugs	burden	camelids
critters(81)	(26|155)=26	(26|6)=3	(26|21)=8	(26|2)=0	(26|2)=2	(26|6)=4	(26|20)=6	(26|14)=2	(26|1)=0	(26|3)=2	(26|4)=1	(26|1)=0
	animals	amphibians	animal	apes	arachnids	arthropods	beasts	birds	bovines	bugs	burden	camelids
vertebrates(7	(30|155)=30	(30|6)=4	(30|21)=12	(30|2)=0	(30|2)=0	(30|6)=1	(30|20)=6	(30|14)=3	(30|1)=0	(30|3)=0	(30|4)=1	(30|1)=0
	animals	amphibians	animal	apes	arachnids	arthropods	beasts	birds	bovines	bugs	burden	camelids
game(76)	(36|155)=36	(36|6)=1	(36|21)=11	(36|2)=0	(36|2)=0	(36|6)=1	(36|20)=9	(36|14)=2	(36|1)=0	(36|3)=0	(36|4)=1	(36|1)=0
	animals	amphibians	animal	apes	arachnids	arthropods	beasts	birds	bovines	bugs	burden	camelids
food(75)	(19|155)=19	(19|6)=2	(19|21)=5	(19|2)=0	(19|2)=1	(19|6)=4	(19|20)=2	(19|14)=5	(19|1)=0	(19|3)=3	(19|4)=1	(19|1)=0
	animals	amphibians	animal	apes	arachnids	arthropods	beasts	birds	bovines	bugs	burden	camelids
herbivores(75	(29|155)=28	(29|6)=1	(29|21)=9	(29|2)=0	(29|2)=0	(29|6)=2	(29|20)=7	(29|14)=3	(29|1)=1	(29|3)=0	(29|4)=3	(29|1)=0
	animals	amphibians	animal	apes	arachnids	arthropods	beasts	birds	bovines	bugs	burden	camelids

Eval #2: Intermediate concepts

- Human evaluation, four annotators

All concepts before Concept Positioning Test

	Animals				People			
	A1	**A2**	**A3**	**A4**	**A1**	**A2**	**A3**	**A4**
Correct	246	243	251	230	239	231	225	221
Borderline	42	26	22	29	12	10	6	4
BasicConcept	2	8	9	2	6	2	9	10
NotConcept	147	160	155	176	39	53	56	61
Acc1 %	0.56	0.56	0.57	0.53	0.81	0.78	0.76	0.75
Acc2 %	0.66	0.62	0.62	0.59	0.85	0.81	0.78	0.76

Good concepts after Concept Positioning Test

	Animals after CPT				People after CPT			
	A1	**A2**	**A3**	**A4**	**A1**	**A2**	**A3**	**A4**
Correct	146	133	144	141	126	126	114	116
Borderline	11	15	9	13	6	2	2	0
BasicConcept	2	8	9	2	0	1	7	7
NotConcept	28	31	25	31	7	10	16	16
Acc1 %	0.78	0.71	0.77	0.75	0.91	0.91	0.82	0.83
Acc2 %	0.84	0.79	0.82	0.82	0.95	0.92	0.83	0.83

Acc1 = percentage *Correct*

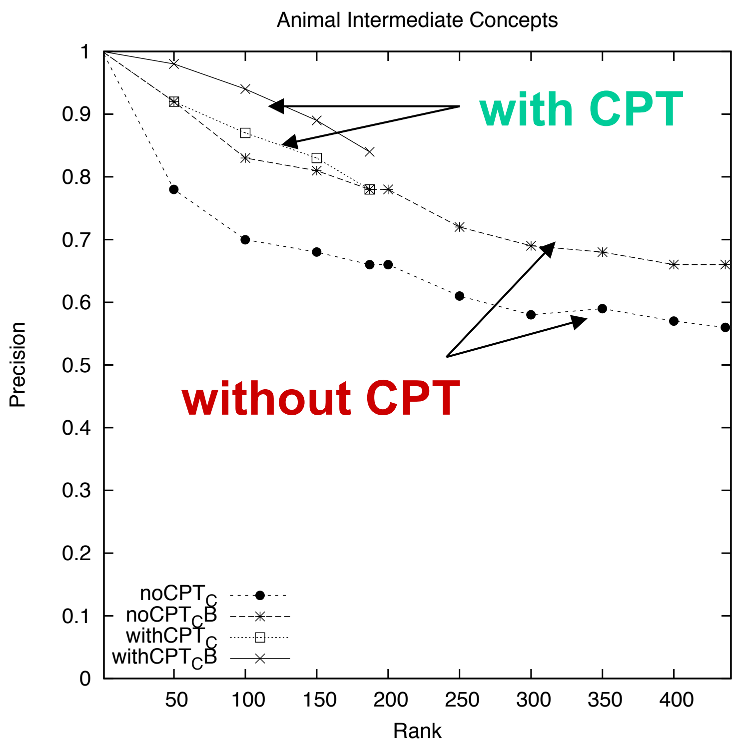
Acc2 = percentage *Correct* or *Borderline*

- Comparison with WordNet

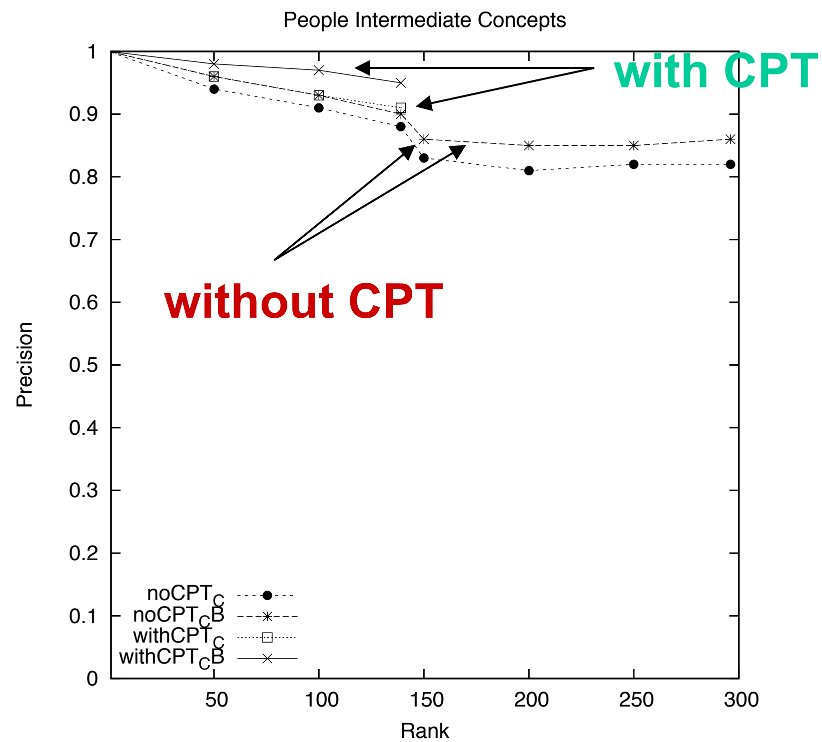
	# harvested	**Pr_{WN}**	**Pr_{HUM}**	**NotInWN**
Animals	**437**	**.20**	**.57**	**204**
People	**296**	**.51**	**.85**	**108**

Effect of In-degree concept ranking

- *In-degree* measures popularity of concept
- Precision drops as In-degree drops:



Animals



People

Evaluation #3: is-a links

- Accuracy of algorithm on taxonomy links?

- Very expensive to consider all links
  - Need concept disambiguation in Wordnet
  - Need manual inspection of each term

- Consider only links from instance/basic level to immediate parent:

	# harvested	Pr_{WN}	Pr_{HUM}	NotInWN
Animals	1940	.47	.88	804
People	908	.23	.94	539

WordNet lacks nearly half of the is-a links!

Human evaluation

- First check if terms are correct:
  - 3 human judges; used web to check
  - Good answer = *Category*; inverse ISA = *Member*; bad term = *Discard*
  - Very high pairwise Cohen kappas

- Then evaluate ISAs:
  - Randomly selected 120 each (Animal and People) relati (100 from harvesting; 20 made at random to include some *False* answers)
  - 3 humans judges; asked if instance *always / sometimes / never* under supercategory
  - Average pairwise Cohen kappa = 0.71 (animals) and 0.84 (people)

Animal				
code	A_D	A_E	A_F	A_G
Category	497	415	396	393
Discard	81	97	82	85
Member	29	25	13	4
Unknown	34	41	20	6
Totals	641	578	511	488
Accuracy	**.78**	**.72**	**.77**	**.81**

People				
code	A_D	A_E	A_F	A_G
Category	349	470	296	260
Discard	35	44	26	46
Member	13	10	10	12
Unknown	19	34	7	4
Totals	416	557	339	322
Accuracy	**.84**	**.84**	**.87**	**.81**

Kappa agreement for Animal					
A_D,A_E	A_D,A_F	A_D,A_G	A_E,A_F	A_E,A_G	A_F,A_G
.74	.83	.76	.78	.83	.82

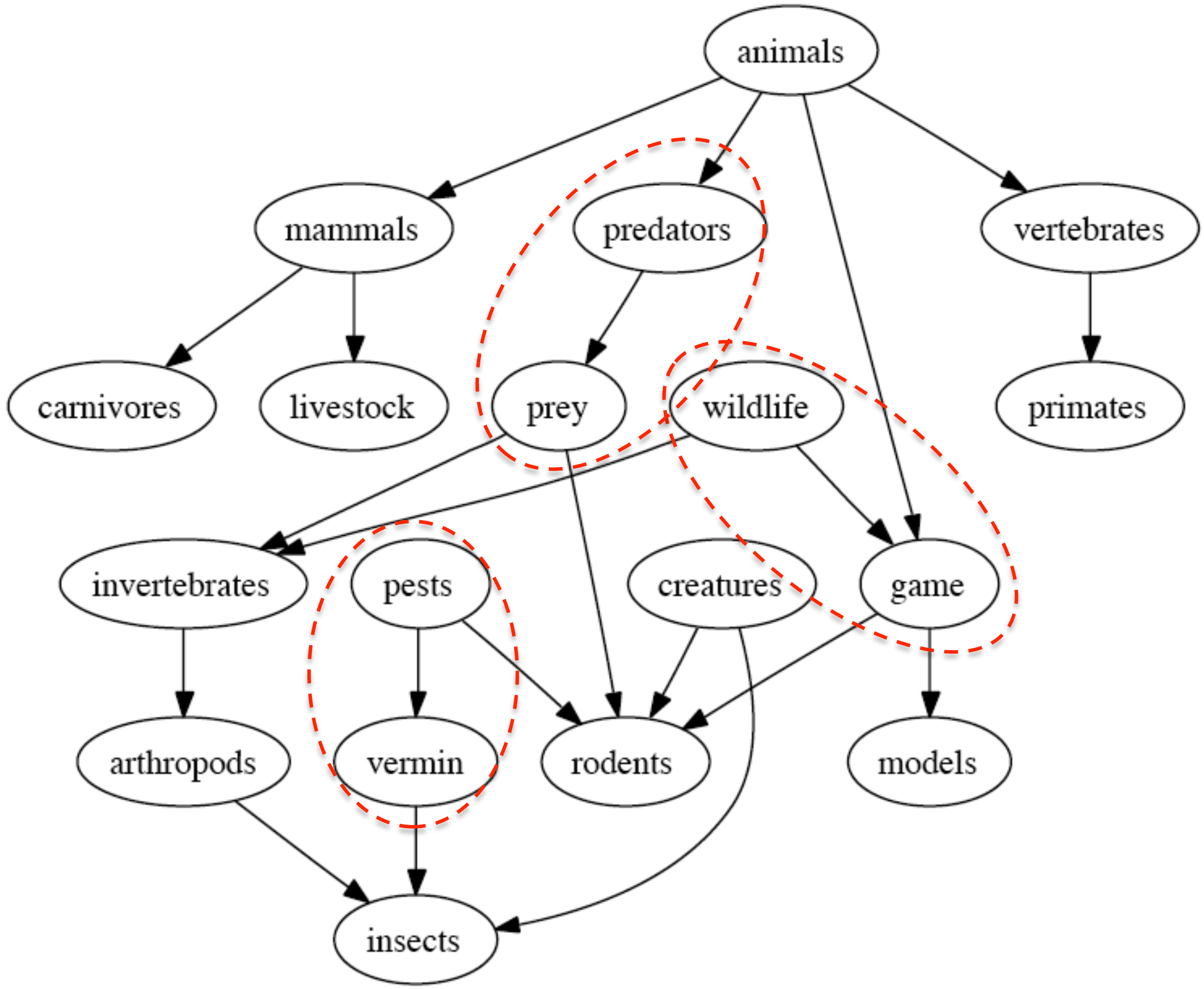
Kappa agreement for People					
A_D,A_E	A_D,A_F	A_D,A_G	A_E,A_F	A_E,A_G	A_F,A_G
.87	.86	.85	.84	.87	.89

	Animals		People	
code	A_B	A_K	A_B	A_K
AlwaysTrue	79	58	82	75
AlwaysFalse	24	23	8	13
Sometimes	12	30	26	23
NoIdea	5	7	4	7
Accuracy	.76	.73	.90	.82

Still…results are a bit of a mess

The problem?

Too many different kinds of categories



Solution: Group classes into small sets

- Goal: Create smaller sets, then taxonomize
- Need to find groups / families of classes

  [*predators prey*]

  [*carnivores herbivores omnivores*]

  [*pets wild_animals lab_animals* …]

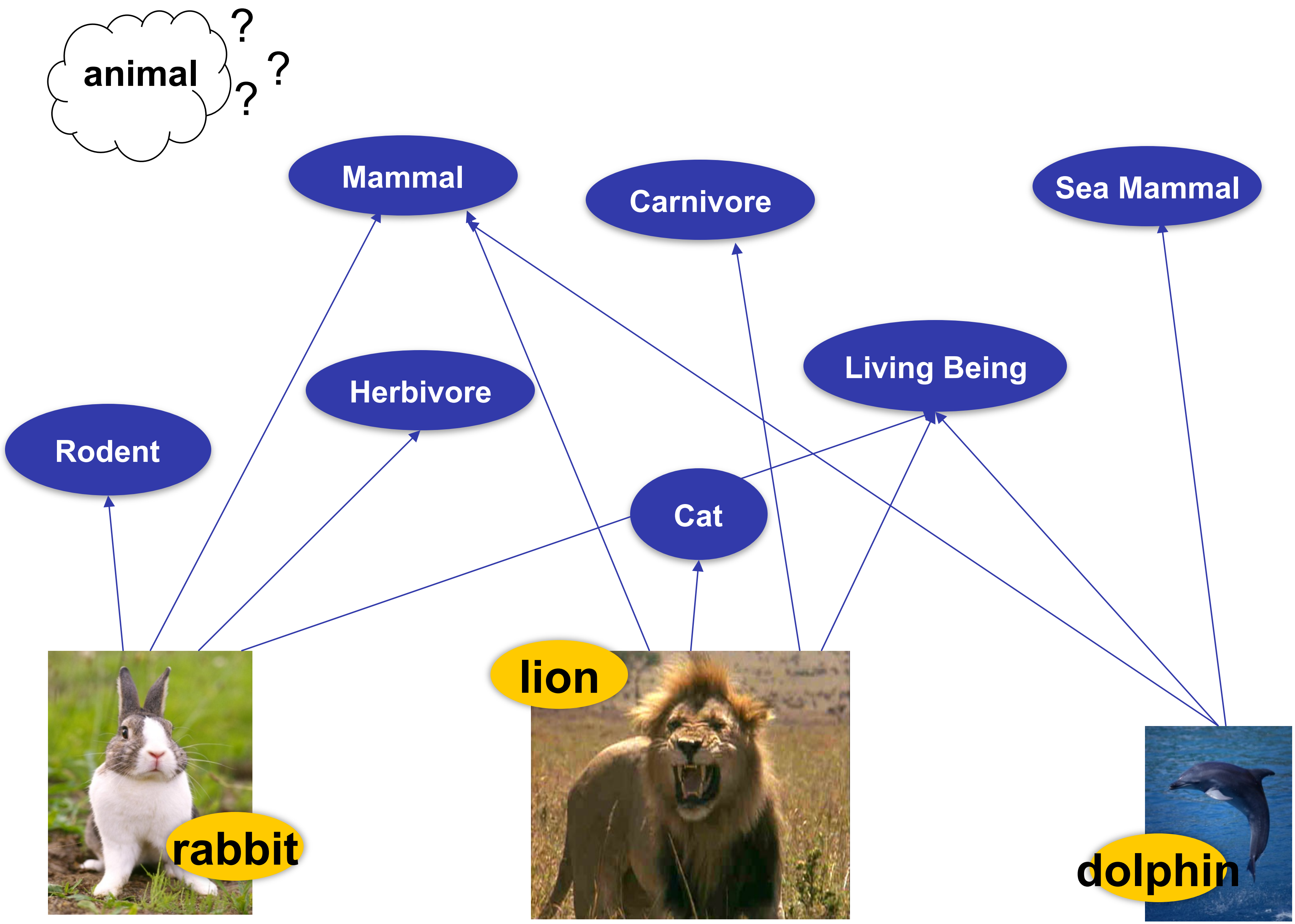
  [*water_animals land_animals* …]

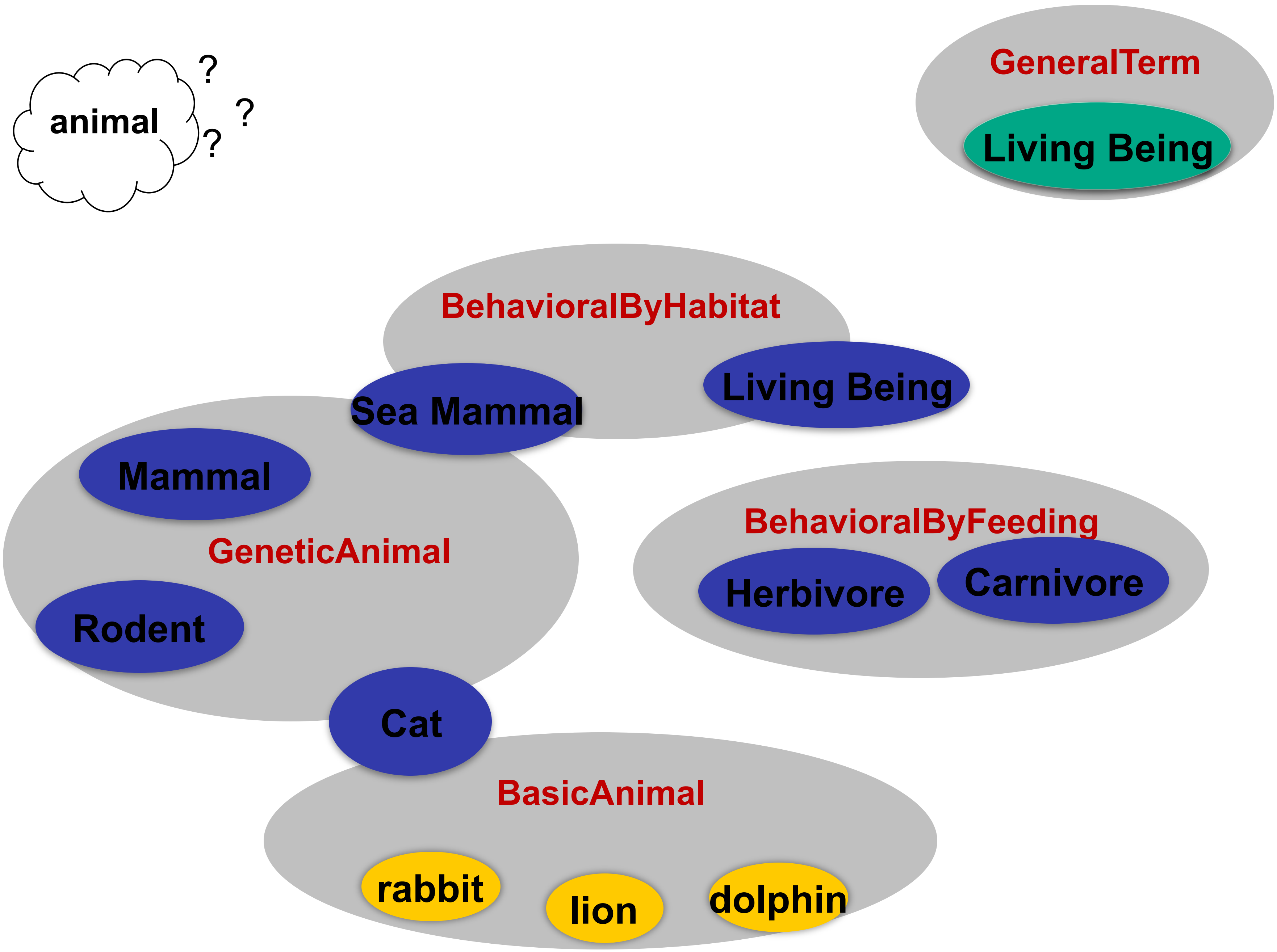
- Approach: Consult online dictionaries, encyclopedias:
  - Some classes are defined by behaviors (such as eating), some by body structure, some by function …
  - Try to define search patterns that capture salient aspects:

    "[*carnivores*|*herbivores*|*omnivores*] *are animals that* ***eat***…"

    "[*water_animals*|*land_animals*] *are animals that* ***live***…"

    "[*pets*|*lab_animals*|*zoo_animals*] *are animals that* ***?*** "

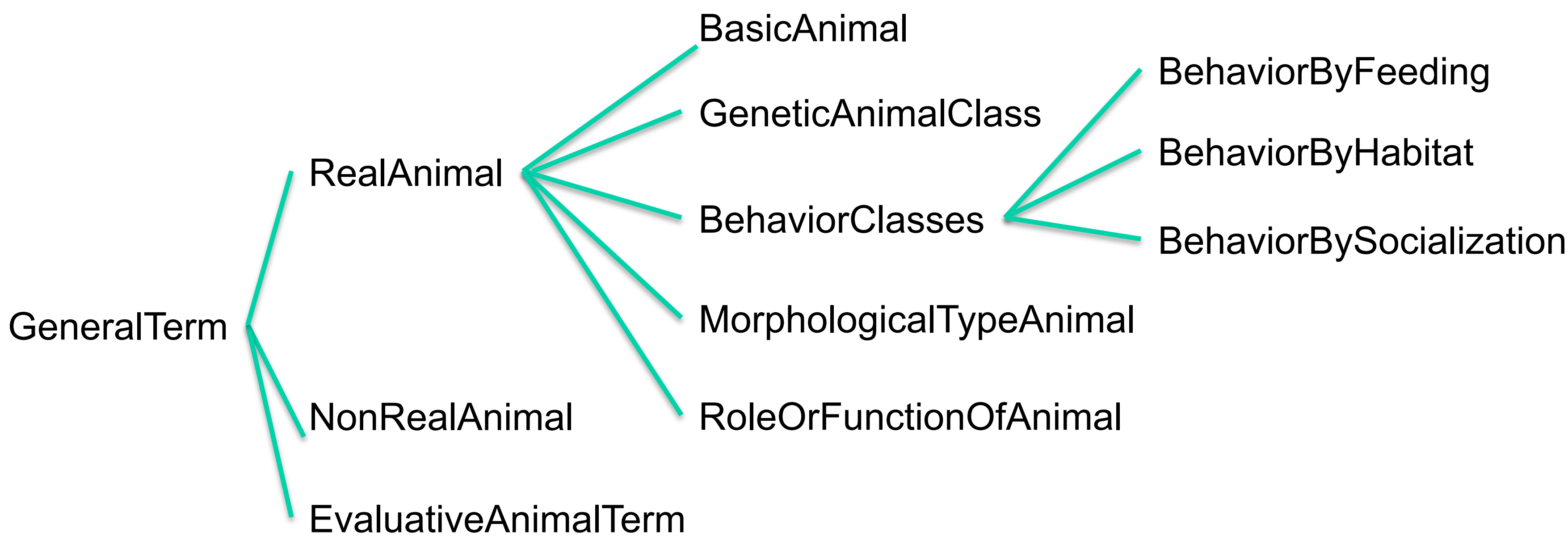

animal
?
?
?
Mammal
Carnivore
Sea Mammal
Herbivore
Living Being
Rodent
Cat
rabbit
lion
dolphin


animal
?
?
?
GeneralTerm
Living Being
BehavioralByHabitat
Sea Mammal
Living Being
Mammal
GeneticAnimal
Rodent
BehavioralByFeeding
Herbivore
Carnivore
Cat
BasicAnimal
rabbit
lion
dolphin

Evaluating sets

(Kozareva et al. AAAI Spring Symp 09)

- First, created a small Upper Model manually:



- Then, had 4 independent annotators choose appropriate Upper Model class(es) for several hundred harvested classes
- Kappa agreement for some classes ok, for others not so good
  - Sometimes quite difficult to determine what an animal term means

1. BasicAnimal

The **basic individual** animal. Can be visualized mentally. Examples: Dog, Snake, Hummingbird.

2. GeneticAnimalClass

A **group** of basic animals, defined by **genetic similarity**. Cannot be visualized as a specific type. Examples: Reptile, Mammal. Note that sometimes a genetic class is also characterized by distinctive behavior, and so should be coded twice, as in Sea-mammal being both GeneticAnimalClass and BehavioralByHabitat. (Since genetic identity is so often expressed as body structure—it's a rare case that two genetically distant things look the same structurally—it will be easy to confuse this class with MorphologicalTypeAnimal. If the term refers to just a portion of the animal, it's probably a MorphologicalTypeAnimal. If you really see the meaning of the term as both genetic and structural, please code both.)

3. NonRealAnimal

Imaginary animals. Examples: Dragon, Unicorn. (Does not include 'normal' animals in literature or films.)

4. BehavioralByFeeding

A type of animal whose essential defining characteristic relates to a **feeding pattern** (either feeding itself, as for Predator or Grazer, or of another feeding on it, as for Prey). Cannot be visualized as an individual animal. Note that since a term like Hunter can refer to a human as well as an animal, it should not be classified as GeneralTerm.

5. BehavioralByHabitat

A type of animal whose essential defining characteristic relates to its habitual or otherwise noteworthy **spatial location**. Cannot be visualized as an individual animal. (When a basic type also is characterized by its spatial home, as in South African gazelle, treat it just as a type of gazelle, i.e., a BasicAnimal. But a class, like South African mammals, belongs here.) Examples: Saltwater mammal, Desert animal. And since a creature's structure is sometimes determined by its habitat, animals can appear as both; for example, South African ruminant is both a BehavioralByHabitat and a MorphologicalTypeAnimal.

6. BehavioralBySocializationIndividual

A type of animal whose essential defining characteristic relates to its patterns of **interaction with other animals**, of the same or a different kind. Excludes patterns of feeding. May be visualized as an individual animal. Examples: Herding animal, Lone wolf. (Note that most animals have some characteristic behavior pattern. So use this category only if the term explicitly focuses on behavior.)

7. BehavioralBySocializationGroup

A natural **group of basic** animals, defined by **interaction with other animals**. Cannot be visualized as an individual animal. Examples: Herd, Pack.

8. MorphologicalTypeAnimal

A type of animal whose essential defining characteristic relates to its internal or external **physical structure** or appearance. Cannot be visualized as an individual animal. (When a basic type also is characterized by its structure, as in Duck-billed platypus, treat it just as a type of platypus, i.e., a BasicAnimal. But a class, like Armored dinosaurs, belongs here.) Examples: Cloven-hoofed animal, Short-hair breed. And since a creature's structure is sometimes determined by its habitat, animals can appear as both; for example, South African ruminant is both a MorphologicalTypeAnimal and a BehavioralByHabitat. Finally, since genetic identity is so often expressed as structure—it's a rare case that two genetically distant things look the same structurally—it will be easy to confuse this class with MorphologicalTypeAnimal. If the term refers to just a portion of the animal, it's probably a MorphologicalTypeAnimal. But if you really see both meanings, please code both.

9. RoleOrFunctionOfAnimal

A type of animal whose essential defining characteristic relates to the **role or function** it plays with respect to others, typically humans. Cannot be visualized as an individual animal. Examples: Zoo animal, Pet, Parasite, Host.

G. GeneralTerm

A term that includes animals (or humans) but refers *also* to things that are neither animal nor human. Typically either a very general word such as Individual or Living being, or a general role or function such as Model or Catalyst. Note that in rare cases a term that refers mostly to animals also includes something else, such as the Venus Fly Trap plant, which is a carnivore. Please ignore such exceptional cases. But when a large proportion of the instances of a class are non-animal, then code it as GeneralTerm.

E. EvaluativeAnimalTerm

A term for an animal that carries an opinion judgment, such as "varmint". Sometimes a term has two senses, one of which is just the animal, and the other is a human plus a connotation. For example, "snake" or "weasel" is either the animal proper or a human who is sneaky; "lamb" the animal proper or a person who is gentle, etc. Since the term can potentially carry a judgment connotation, please code it here as well as where it belongs.

A. OtherAnimal

Almost certainly an animal or human, but none of the above applies, or: "I simply don't know enough about it".

Taxonomization evaluation 1: Animals

Human Judgement Animal

	An1	An2	An3	An4	K
BasicAnimal	29	24	13	4	.51
BehByFeeding	48	33	45	49	.68
BehByHabitat	85	58	56	54	.66
BehBySocGroup	1	2	6	7	.47
BehBySocInd	5	4	1	0	.46
EvaluativeTerm	41	14	10	29	.51
Garbage Term	21	12	15	16	.74
GeneralTerm	83	72	64	79	.52
GeneticAnimal	95	113	81	73	.61
MorphTypeAnimal	29	33	42	39	.58
NonRealAnimal	0	1	0	0	.50
NotAnimal	81	97	82	85	.68
OtherAnimal	34	41	20	6	.47
Role/FunctAnimal	89	74	76	47	.58
Total	641	578	511	488	.57

Class definition

[illegible]

Taxonomization evaluation 2: People

Human Judgement People

	An1	An2	An3	An4	K
BasicPerson	5	6	1	3	.55
FamilyRelation	7	6	7	6	.86
GeneralTerm	38	12	21	12	.50
GeneticPersonCl	1	2	1	0	.44
ImaginaryPeople	14	16	5	2	.47
NationOrTribe	2	3	3	2	.78
NonTranEventPar	29	63	41	32	.57
NotPerson	31	31	28	38	.80
OtherHuman	4	5	0	2	.50
PersonState	23	1	25	1	.47
RealPeople	1	7	1	0	.50
ReligiousAffiliation	10	16	12	15	.61
SocialRole	62	61	39	44	.61
TransientEventPar	30	27	13	7	.48
Total	257	256	197	164	.58

Class definition

NonTransientEventParticipant

The **role a person plays consistently over time, by taking part** in an **action or activity that either persists or recurs**, without a specific end-point being defined.

The group includes several types: Occupations (priest, doctor), Hobbies (skier, collector), Habits (stutter, peacemaker).

TransientEventParticipant

The **role a person plays for a limited time**, through taking part in one or more specific well-defined events. This class is distinguished as a **characteristic action or activity**.

Example: speaker, passenger, visitor.

PersonState

A person with a **certain physical or mental characteristic that persists over time**. Distinguishing this class from NonTransientEventParticipant, there is **no typical associated defining action** or activity that one can think of.

Example: schizophrenic, AIDS patient, blind person.

Code	An1	An2	An3	An4	Ex.M	Par.M	Kappa
BasicAnimal	29	24	13	4	2	12	0.51
BehavioralByFeeding	48	33	45	49	27	17	0.68
BehavioralByHabitat	85	58	56	54	36	36	0.66
BehavioralBySocializationGroup	1	2	6	7	0	3	0.47
BehavioralBySocializationIndividual	5	4	1	0	0	2	0.46
EvaluativeTerm	41	14	10	29	6	19	0.51
GarbageTerm	21	12	15	16	12	3	0.74
GeneralTerm	83	72	64	79	19	72	0.52
GeneticAnimalClass	95	113	81	73	42	65	0.61
MorphologicalTypeAnimal	29	33	42	39	13	26	0.58
NonRealAnimal	0	1	0	0	0	0	0.50
NotAnimal	81	97	82	85	53	40	0.68
OtherAnimal	34	41	20	6	1	24	0.47
RoleOrFunctionOfAnimal	89	74	76	47	28	56	0.58
Totals	**641**	**578**	**511**	**488**	**239**	**375**	**0.57**

Human category judgments

Animals

People

Code	An1	An2	An3	An4	Ex.M	Par.M	Kappa
BasicPerson	5	6	1	3	1	3	0.55
FamilyRelation	7	6	7	6	5	2	0.86
GeneralTerm	38	12	21	12	4	18	0.50
GeneticPersonClass	1	2	1	0	0	1	0.44
ImaginaryPeople	14	16	5	2	1	10	0.47
NationOrTribe	2	3	3	2	2	1	0.78
NonTransientEventParticipant	29	63	41	32	16	33	0.57
NotPerson	31	31	28	38	24	9	0.80
OtherHuman	4	5	0	2	0	0	0.50
PersonState	23	1	25	1	0	8	0.47
RealPeople	1	7	1	0	0	1	0.50
ReligiousAffiliation	10	16	12	15	5	11	0.61
SocialRole	62	61	39	44	25	36	0.61
TransientEventParticipant	30	27	13	7	2	17	0.48
Totals	**257**	**256**	**197**	**164**	**85**	**150**	**0.58**

Simplifying intermediate classes

- Agreement still low…
- So: Grouped sets into 4 categories
- Used same 4 humans
- Pairwise inter-annotator agreement (Fleiss kappa, Fleiss 71):
  – Animals 0.61–0.71 (avg 0.66)
  – People 0.51–0.70 (avg 0.60)

ANIMAL

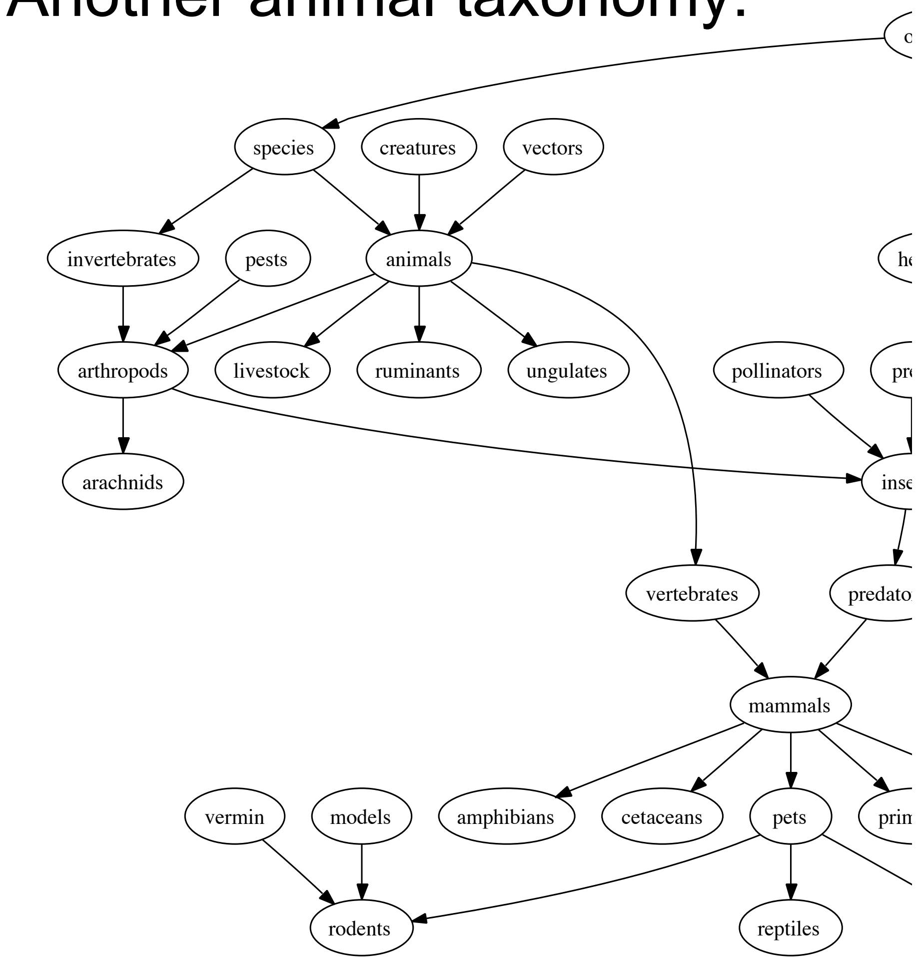
TYPE	LABEL	EXAMPLES
Correct	GeneticAnimal	*reptile,mammal*
	BehavioralByFeeding	*predator, grazer*
	BehaviorByHabitat	*saltwater mammal*
	BehaviorSocialIndiv	*herding animal*
	BehaviorSocialGroup	*herd, pack*
	MorphologicalType	*cloven-hoofed animal*
	RoleOrFunction	*pet, parasite*
Borderline	NonRealAnimal	*dragons*
	EvaluativeTerm	*varmint, fox*
	OtherAnimal	*critter, fossil*
BasicConcept	BasicAnimal	*dog, hummingbird*
NotConcept	GeneralTerm	*model, catalyst*
	NotAnimal	*topic, favorite*
	GarbageTerm	*brates, mals*

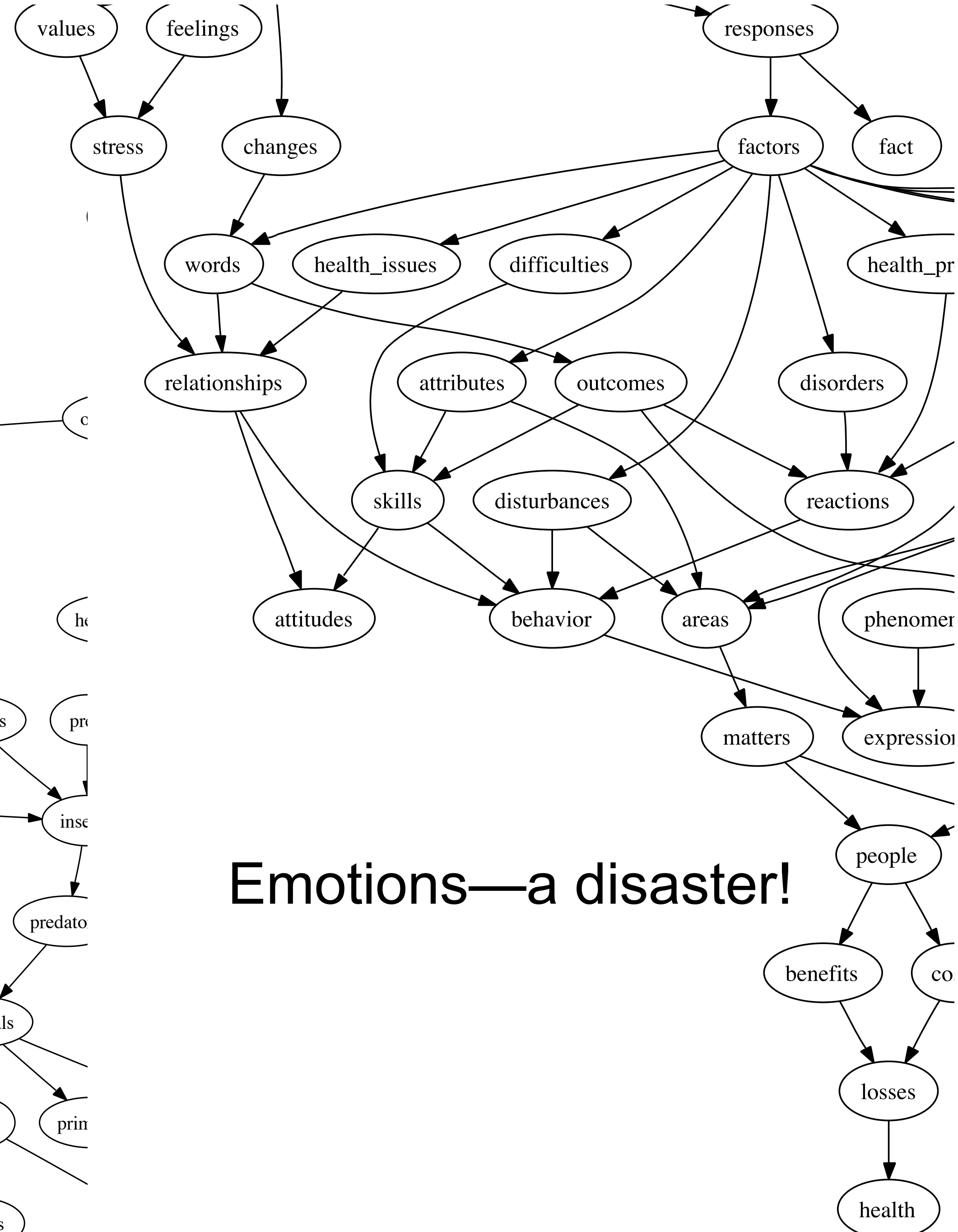
PEOPLE

TYPE	LABEL	EXAMPLES
Correct	GeneticPerson	*Caucasian, Saxon*
	NonTransientEventRole	*stutterer, gourmand*
	TransientEventRole	*passenger, visitor*
	PersonState	*dwarf, schizophrenic*
	FamilyRelation	*aunt, mother*
	SocialRole	*fugitive, hero*
	NationOrTribe	*Bulgarian, Zulu*
	ReligiousAffiliation	*Catholic, atheist*
Borderline	NonRealPerson	*biblical figures*
	OtherPerson	*colleagues, couples*
BasicConcept	BasicPerson	*child, woman*
	RealPerson	*Barack Obama*
NotConcept	GeneralTerm	*image, figure*
	NotPerson	*books, events*

More taxonomies…
still not so great…

Another animal taxonomy:





Emotions—a disaster!

Discussion

- Evaluation is very difficult:
  - Sometimes it is quite difficult to determine what a concept means
  - No standardized and complete and correct resource
  - Unclear precisely what ‘correct’ is-a is
  - What about multiclass assignment?
  - Term space keeps growing and changing
  - Fleiss / Kappa agreements are good for some cases and not so good for others
- But the task is not hopeless!
  - Instance learning is very promising using other forms of DAP or new doubly-anchored patterns, e.g., [NP1 and * and other NP0s]
  - Decomposing ISA structure into small local taxonomies with appropriate sets of intermediate concepts is a way to go

Conclusions regarding DAP

- All experiments are conducted with **DAP and DAP^{-1}**: doubly-anchored pattern starting only with one class name and one class member, or two members

- **DAP is simple, yet very powerful**: harvests knowledge and positions learned concepts

- The bootstrapping **algorithm serves multiple purposes**:
  - generates highly accurate, rich and diverse lists of concepts
  - finds instances and intermediate concepts that are missing from WordNet
  - learns partial taxonomic structures

- **Category evaluation is challenging even for humans**, because it is difficult to determine the meaning of a concept

Part 6

LEARNING RELATIONS

Argument harvesting (Kozareva and Hovy EMNLP 10)

- Use a recursive DAP pattern that starts with a target **relation** and one **seed** argument and learns new arguments
- Submit query to Yahoo!

Mary
Peter
Emma

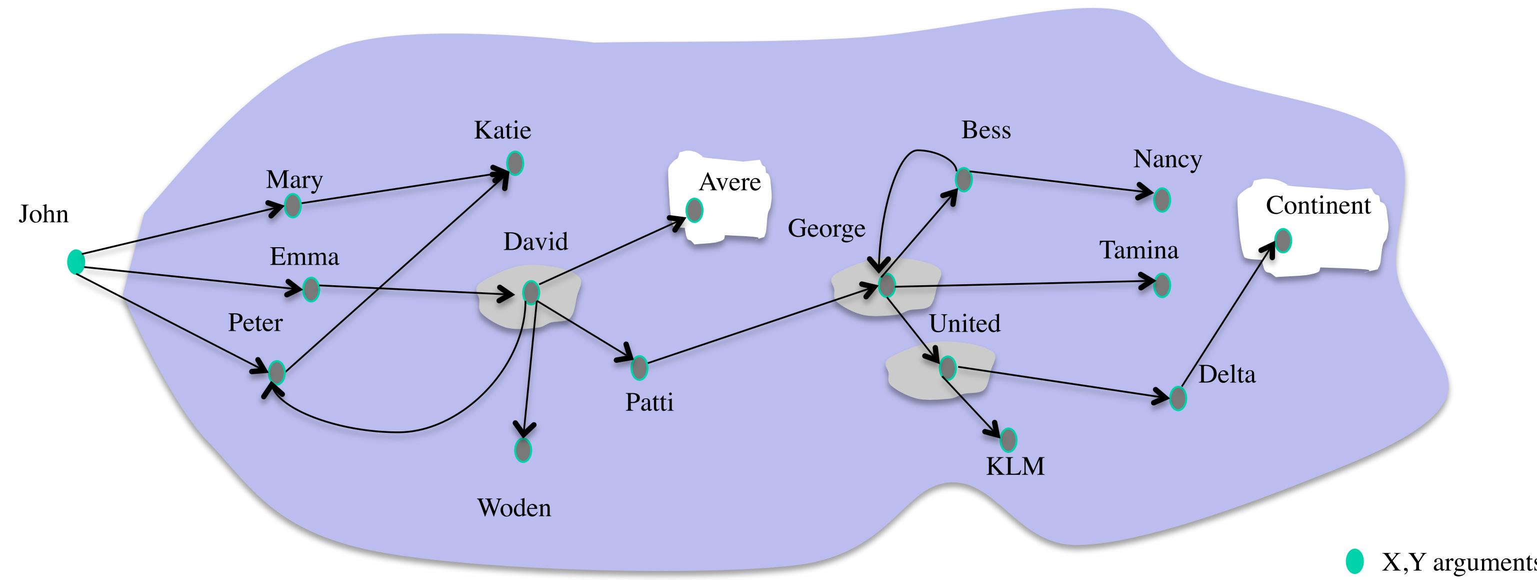
and **John** [fly to]

New York
Italy
party

- Run an exhaustive breadth-first search
- In each iteration, add only unexplored instances to the query queue

Argument ranking: Y elements

- Build a directed graph using the ***X and Y fly to***



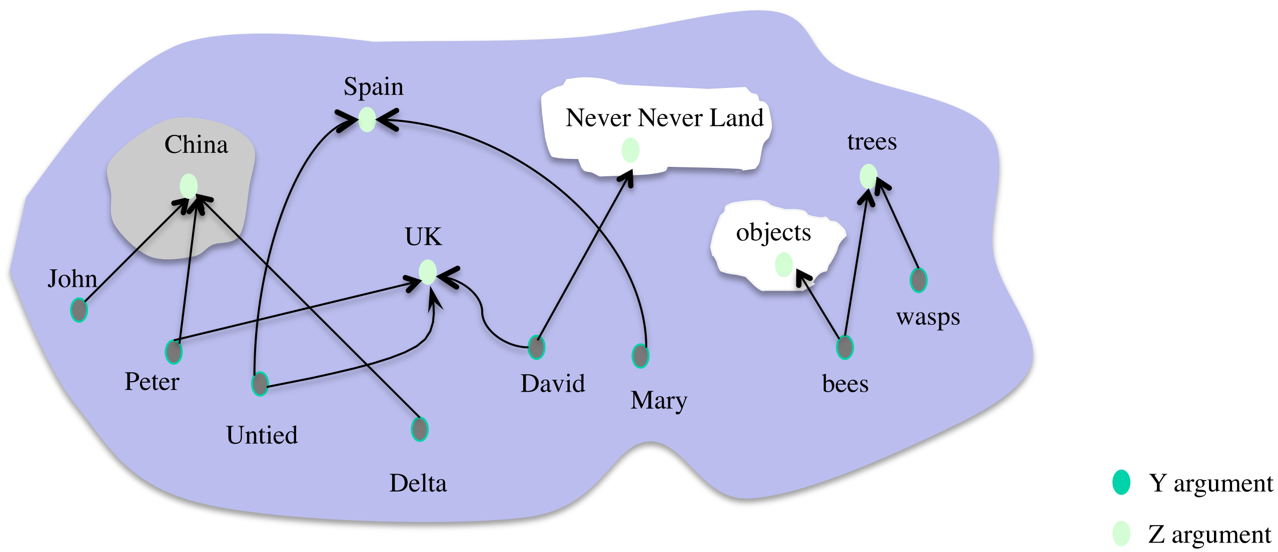
- Rank elements
  - totalDegree of a node (v) is

$$totD(v) = \frac{\sum_{v,u \in E} w(v,u) + \sum_{u,v \in E} w(u,v)}{|V-1|}$$

  the sum of all outgoing and incoming edges from v normalized by V-1

Argument ranking: Z elements

- Build a directed graph using the ***Y fly to Z***



- Rank Z elements
  - <u>inDegree</u> of a node (*v'*) is $$inD(v') = \frac{\sum_{u',v' \in E'} w(u',v')}{|V'-1|}$$

    the sum of all incoming edges from y arguments *u'* towards *v'* normalized by V'-1

Supertype harvesting

- Next apply supertype DAP pattern (Hovy et al., 2009)

  “ * such as <argument1> and <argument2> “

- Submit query to Yahoo!

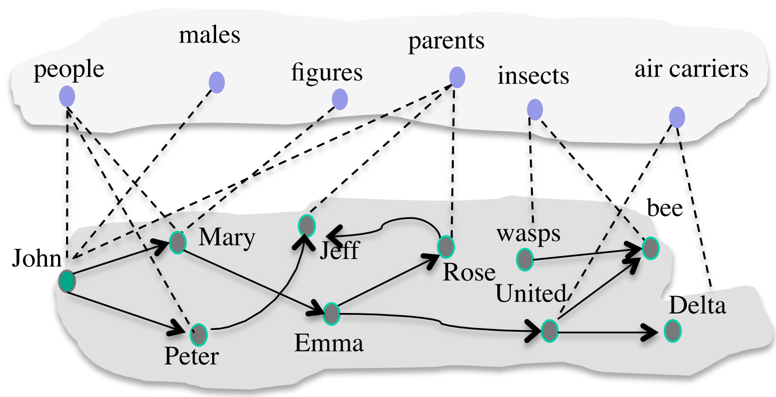
people
individuals
airlines
carriers
…

such as

Mary and **John**
Peter and **John**
Emma and **John**
…
Delta and **United**
Delta and **American**
KLM and **Alitalia**

Supertype ranking

- Build a directed graph of ***Yarg-Zarg-supertype*** triples



- Rank elements
  - inDegree of a supertype node (*v"*) is the sum of all incoming edges from the argument pairs towards *v"* normalized by V"-1

Experiment: 14 relations

Harvesting Procedure:

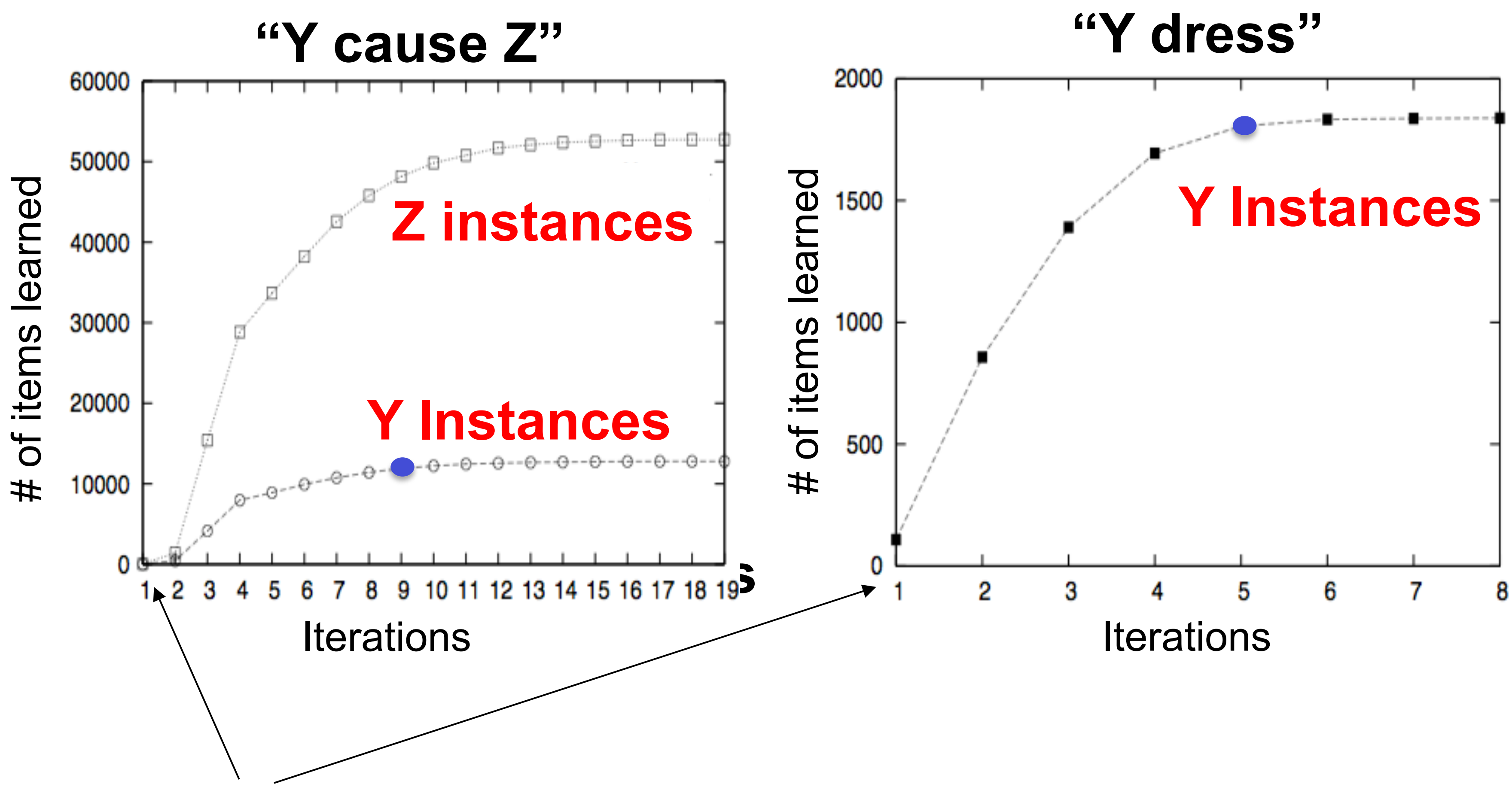
- submit patterns as Web queries
- collect 1000 snippets per query
- keep only unique answers
- run bootstrapping until exhaustion

- harvested 30GB of data
- learned 189,090 terms for 14 relations

- wide number diversity

Lexico-Syntactic Pattern	#Iterations	#Y arg.	#Z arg.
* and Easyjet fly to *	19	772	1176
* and Rita go to *	13	18406	27721
* and Charlie work for *	20	2949	3396
* and Scott work at *	15	1084	1186
* and Mary work on *	7	4126	5186
* and John work in *	13	4142	4918
* and Peter live with *	11	1344	834
* and Donald live at *	15	1102	1175
* and Harry live in *	15	8886	19698
* and virus cause *	19	12790	52744
* and Jim celebrate	12	6033	-
* and Sam drink	13	1810	-
* and scared people	17	2984	-
* and nice dress	8	1838	-

Learning curves



Baseline: terms harvested with singly-anchored patterns

Good iteration stopping points

Evaluation problems

- What to compare results to?
- Most approaches
  - do not learn the supertypes of the arguments
  - map the information to existing repository like WordNet (Pantel and Pennacchiotti, 2006)
- The point of our work is to learn more/new terms than are currently available:
  - compare against an existing repository
  - conduct manual evaluation of top ranked arguments and supertypes

Evaluation #1 by humans: Arguments

- Human evaluation of top 200 arguments for all fourteen relations
- When the algorithm claims that (*X relation Z*)
  - (1) is it true that *X* and *Z* are correct fillers?
  - (2) of what type?

X WorkFor	A1	A2	WorkFor Z	A1	A2
Person	148	152	Organization	111	110
Role	5	7	Person	60	60
Group	12	14	Time	4	5
Organization	8	7	Event	4	2
NonPhysical	22	23	NonPhysical	18	19
Error	5	5	Error	3	4
Accuracy	.98	.98	Accuracy	.98	.98

Ron, Kelly

senators, team

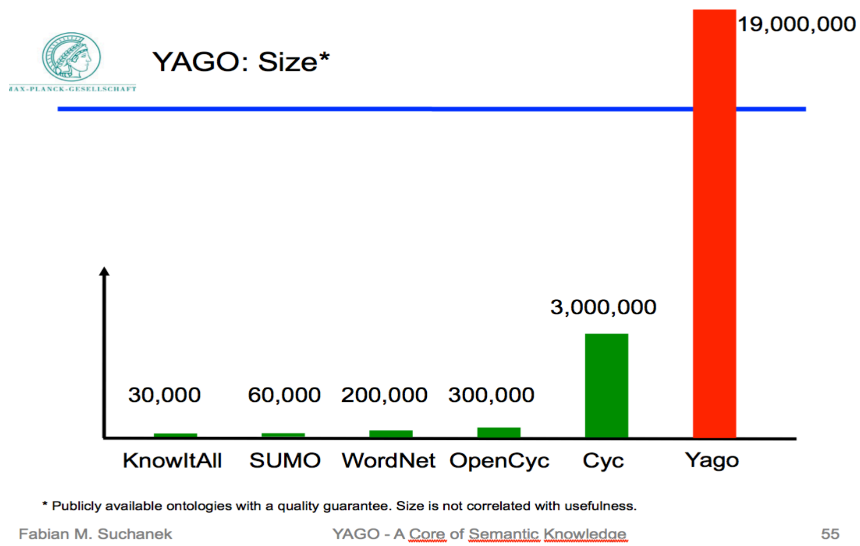
pharmaceutical company

party, prom

glory, fun

Comparison with Yago (Suchanek et al.)

- Yago is much larger than anything else:



  – Majority of the harvested relations are not present

    celebrate, people, dress, drink, cause, liveAt
    liveWith, workOn, workFor, workIn, goTo, flyTo

  – For those found in Yago (*liveIn* and *workAt*), many of the learned terms are missing even though they are sensible and potentially valuable

Evaluating arguments with Yago, 1

- Comparison with Yago

	# harvested	inYago	PrYago	PrHum
X LiveIn	8886	14705	.19	.58
LiveIn Z	19698	4754	.10	.72
X WorkAt	1084	1399	.12	.88
WorkAt Z	1186	525	.3	.95

$$\Pr Yago = \frac{\#terms\_found\_in\_Yago}{\#terms\_harvested\_by\_system}$$

$$\Pr Hum = \frac{\#terms\_judged\_correct\_by\_human}{\#terms\_harvested\_by\_system}$$

Evaluating arguments with Yago, 2

- Comparison with Yago

	# harvested	inYago	PrYago	PrHum	NotInYago
X LiveIn	8886	14705	.19	.58	2302
LiveIn Z	19698	4754	.10	.72	13753
X WorkAt	1084	1399	.12	.88	792
WorkAt Z	1186	525	.3	.95	1113

Yago lacks nearly half of the X,Z arguments!

found in both systems

Person names
Locations:
- country (Italy, France, …)
- city (New York, Boston, …)

Institutions:
- universities

NotInYago

Manner of living:
- pain, effort, ease

Locations:
- slums, box, desert

Companies:
- law firm, Microsoft, Starbucks

Research Centers: CERN, Ford

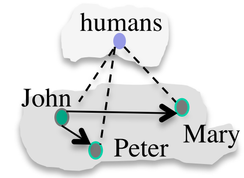
Error analysis

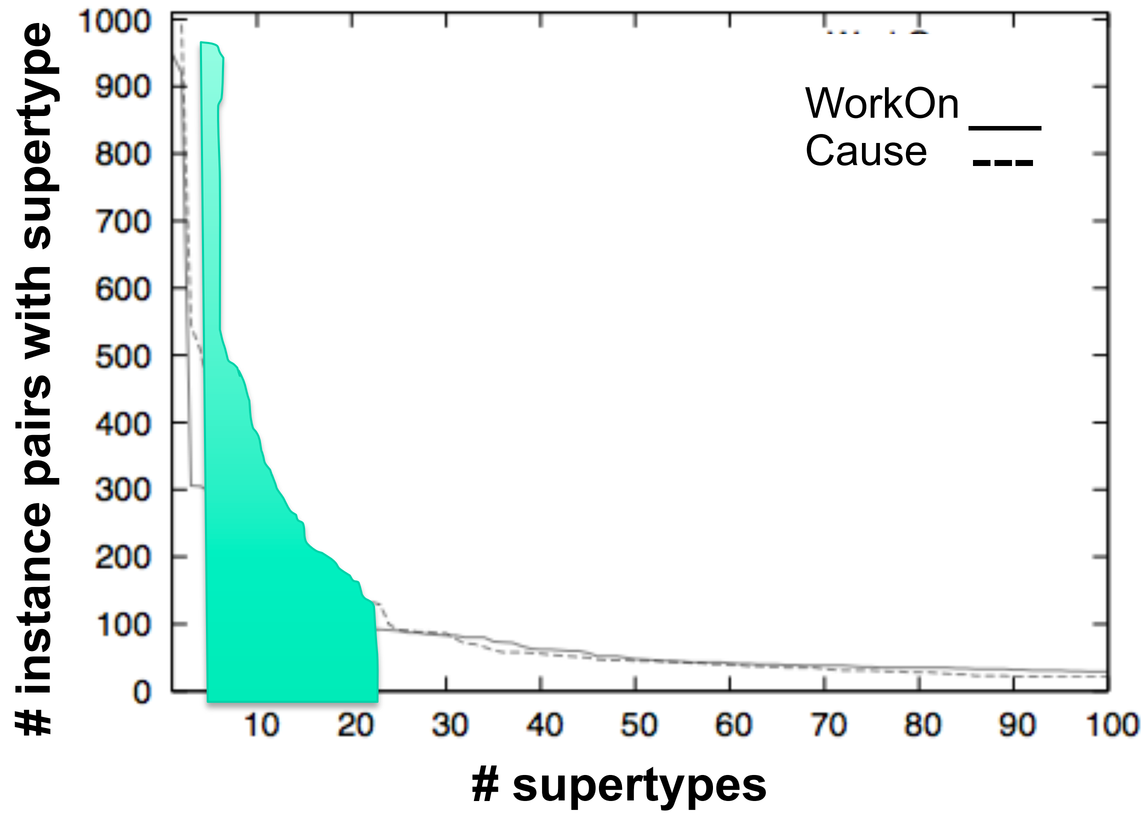
- Type 1: part-of-speech tagging
  - *Cat, [Squirrel]PN and [Duck]PN live in an old white cabin deep in the woods.*
  - *Blank And Jones – [Live]VBP In The Mix (N-Joy)-02-28-CABLE-2004-QMI (. 79.92 MiB. Music. 07/15/04*

- Type 2: fact extraction from fiction books, movie cites, blogs and forums
  - *Fans of the film will know that Sulley and Mike work for [Monsters, Inc.], a power company with a difference — they generate all their power from children's…*

- Type 3: incomplete snippets

Evaluation #2: Supertypes





- The text on the Web prefers a small set of supertypes
- The most popular supertypes are the most descriptive terms

Examples of learned supertypes

Relation	Supertypes
(Sup_x) Dress:	colors, effects, color tones, activities, pattern, styles, material, size, languages, aspects
(Sup_x) FlyTo:	airlines, carriers, companies, giants, people, competitors, political figures, stars, celebs
Cause (Sup_z):	diseases, abnormalities, disasters, processes, issues, disorders, discomforts, emotions, defects, symptoms
WorkFor (Sup_z):	organizations, industries, people, markets, men, automakers, countries, departments, artists, media

Summary

- Automated procedure to learn the selectional restrictions (arguments and supertypes) of semantic relations from the Web
  - finds richer and diverse lists of terms missing from existing knowledge base
  - taxonomizes the arguments linking them with supertypes

Summary

- Novel representation of semantic relations using recursive patterns
- All experiments are conducted with one lexico-syntactic pattern and one seed example
- Recursive patterns are simple and yet very powerful:
  - extract high quality non-trivial information from unstructured text
  - achieve higher recall than singly-anchored ones

CONCLUSION

Tons of related work

- Hyponym and hypernym learning (Hearst 92; Pasca 04, Etzioni et al. 05; Kozareva et al. 08)
- Learning semantic relations (Berland and Charniak 99; Ravichandran and Hovy, 02; Girju et al. 03; Davidov et al. 07)
- Automatic ontology construction (Caraballo 99; Cimiano and Volker 05; Mann 05; Mitchell et al. 2010)
- Usage of lexico-syntactic patterns (Riloff and Jones 99; Fleischman and Hovy 02)
- Unsupervised semantic clustering (Lin 98; Lin and Pantel 02; Davidov and Rapoport 06; Snow and Jurafsky 08)
- Mining knowledge from Wikipedia, e.g. Yago (Suchanek et al. 07)

Future work

- Improve category harvesting and ranking module
- Automatically learn detailed category structure and organize hypernym concepts
- Generate attributes for instances and categories

  …

- Construct ontologies with minimal or almost no supervision

There’s so much to be done

- Learning **inter-concept relations** and their restrictions (parts, attributes, etc.)
- Learning useful and intuitive **taxonomic ‘families’** automatically
- Determining **trustworthiness** of source data
- Handling **change over time**
- Using **multi-linguality** to learn more
- Developing good **evaluation** metrics (Recall of *what* precisely?)

Summary

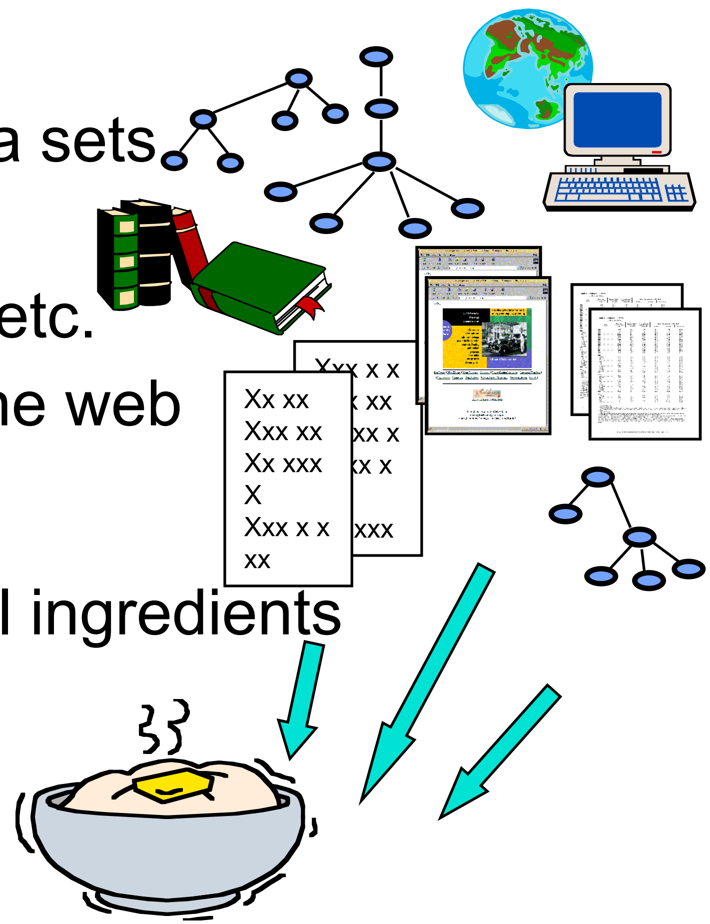
Ingredients:

– small ontologies and metadata sets

– concept families (signatures)

– information from dictionaries, etc.

– additional info from text and the web



Method:

1. Into a large database, pour all ingredients
2. Stir together in the right way
3. Bake

Evaluate—IR, QA, MT, and so on!

Thank you!