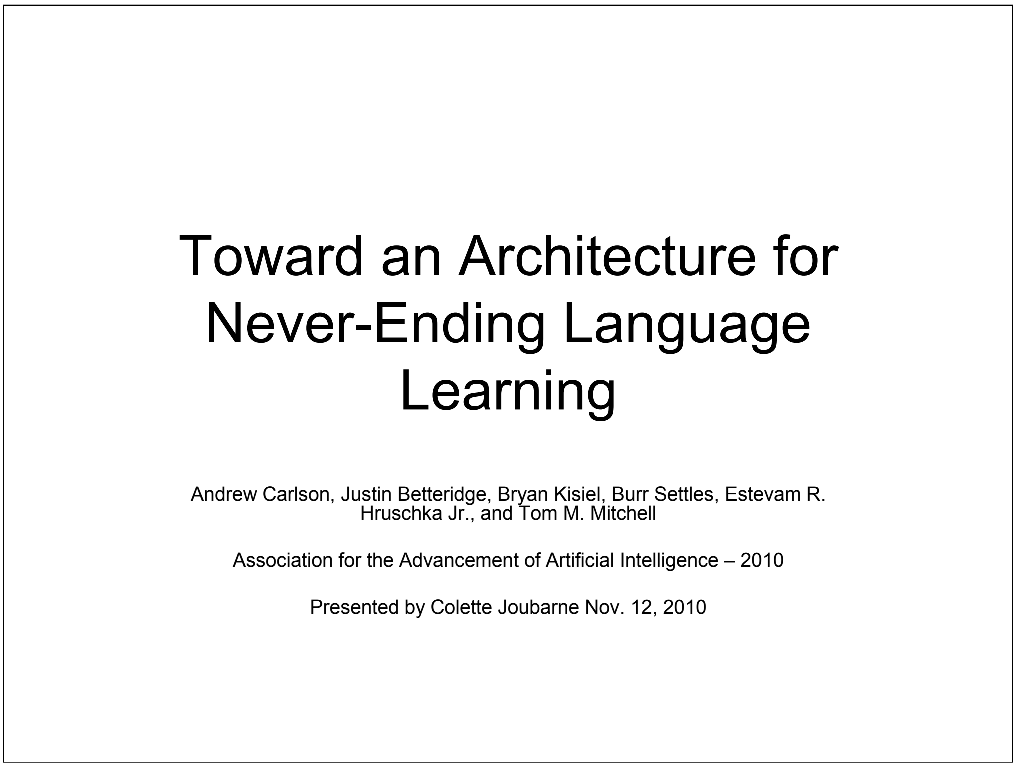
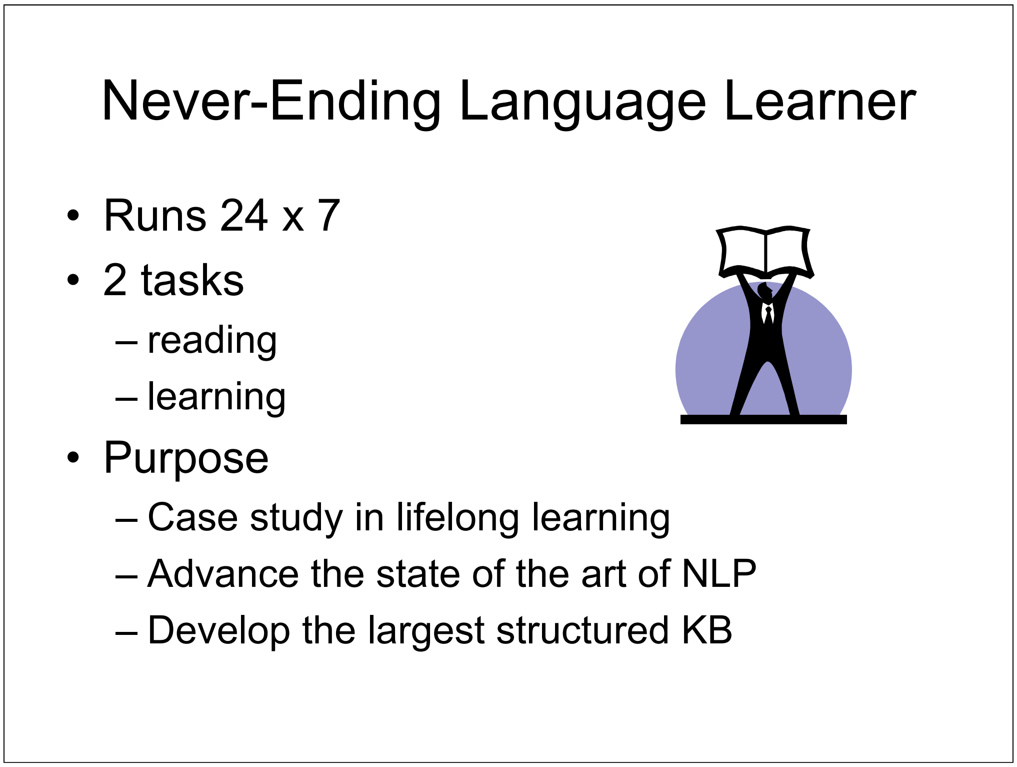


Toward an Architecture for Never-Ending Language Learning

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell

Association for the Advancement of Artificial Intelligence – 2010

Presented by Colette Joubarne Nov. 12, 2010



Never-Ending Language Learner

- Runs 24 x 7
- 2 tasks
  - reading
  - learning
- Purpose
  - Case study in lifelong learning
  - Advance the state of the art of NLP
  - Develop the largest structured KB

-One of many steps towards longer-term goal

-Computer system running 24 x 7

Perform:

-Reading task – extract info from web text to populate a growing KB of structured facts and knowledge

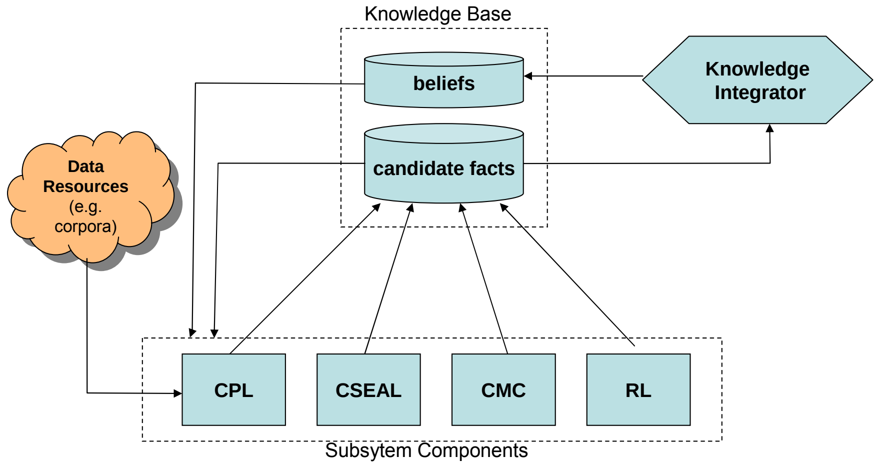
-Learning task – learn to read better each day as evidenced by going back to yesterday's text sources and extracting more info more accurately

-Theory underlying the research is that the vast redundancy of the web (many facts stated many times in diff. ways) will enable a correctly designed system to succeed.

-Couple of Views

> -lifelong = never-ending
>
> -Attempt to advance
>
> -A KB that reflects the factual content of the web, would be useful to many AI efforts

Approach



-Coupled, semi-supervised learning method in which multiple components learn and share complementary types of knowledge, overseen by KI

-Knowledge Base

-grown and used by a collection of learning/reading subsystem components

-initialized with a collection of predicates defining categories and relations, and a handful of seed examples for each predicate

-divided into candidate facts and beliefs

-Subsystem components

-implement complementary knowledge extraction methods

-read from the DB and consult other external resources (corpora or the internet) and propose new candidate facts

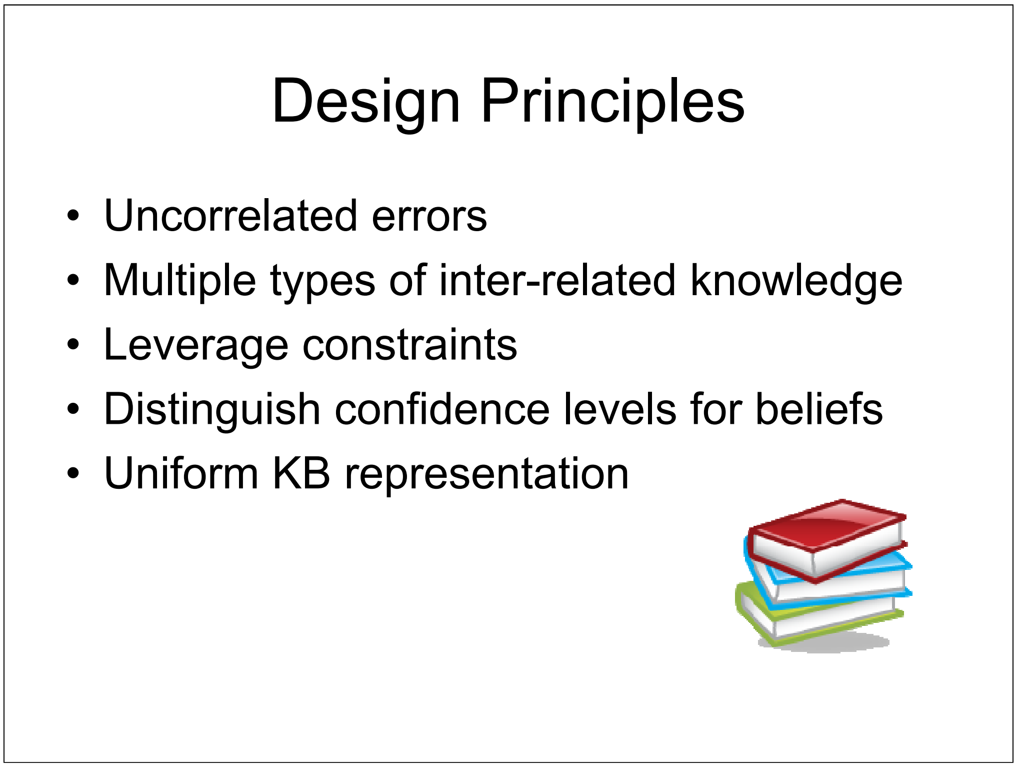
-Supply probability and summary of source evidence supporting each proposed candidate fact

-Knowledge integrator

-Examines proposed candidate facts and promotes the most strongly supported to belief status

-Iterative loop, for each iteration, each subsystem component runs to completion given the current KB, KI makes its decisions.

-This kind of iterative learning approach can result in the accumulation of labelling errors, to help the system stay on track, human interaction will occur 10-15 minutes a day.



Design Principles

- Uncorrelated errors
- Multiple types of inter-related knowledge
- Leverage constraints
- Distinguish confidence levels for beliefs
- Uniform KB representation

-If subsystem component errors are uncorrelated, the probability they all make the same error is the product of their individual error probabilities

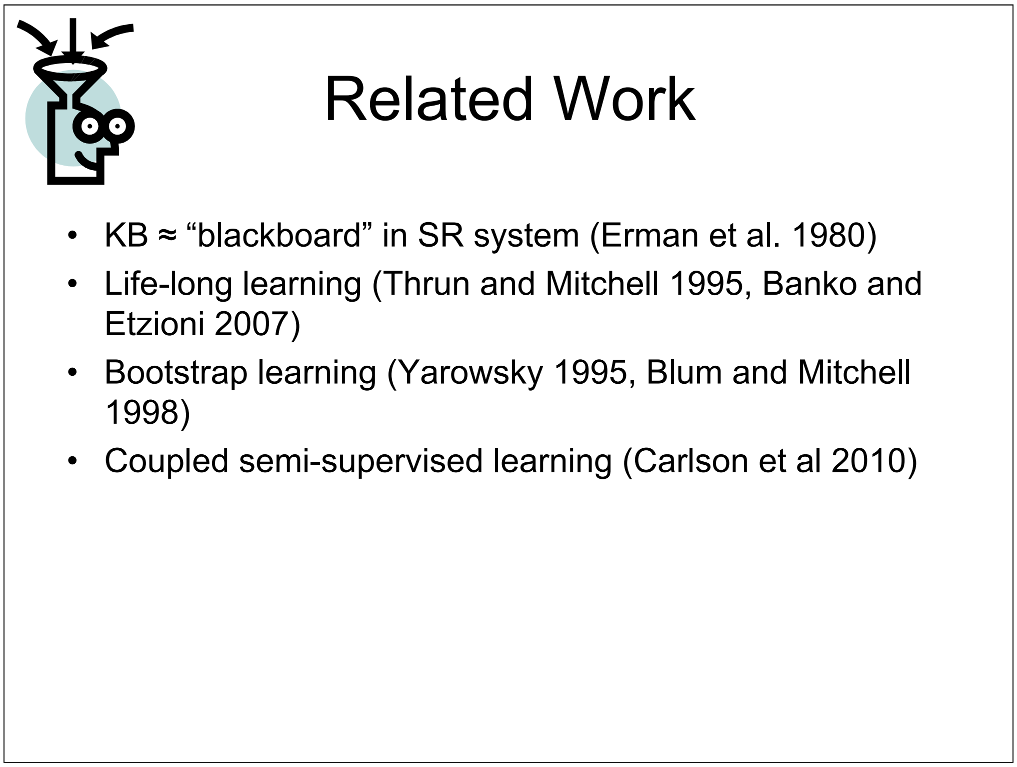
-If each component learns differently, then we have multiple independent sources of the same types of beliefs

> -One component learns to extract predicate instances from text resources and another learns to infer relation instances from other beilefs in the KB

-Arrange categories and relations into a hierarchy that defines which categories are subsets and which are mutually exclusive, and for each relation specify the category of each argument to enable type-checking

-Distinguish high-confidence and low-confidence beliefs in the DB, and retain source justification

-Capture facts and beliefs using a uniform KB representation, so that inference and learning mechanisms can operate on this common representation



Related Work

- KB ≈ “blackboard” in SR system (Erman et al. 1980)
- Life-long learning (Thrun and Mitchell 1995, Banko and Etzioni 2007)
- Bootstrap learning (Yarowsky 1995, Blum and Mitchell 1998)
- Coupled semi-supervised learning (Carlson et al 2010)

Blackboard:

-In 1980 Erman et al characterized the speech-understanding problem as attempting to find a solution using highly diverse and uncertain knowledge. To avoid a combinatorial explosion, they proposed that a powerful control scheme is required to exploit selectively the most promising combinations of alternatives. For example in the domain of SR, initially a word can be any word, but once the word has been identified as an adjective, it highly likely that the next word will be a noun or another adjective.

-Erman et al, implemented their SR system as a collection of key functions performed by independent knowledge sources (or components) which communicated through a global database called the “blackboard”. The blackboard records the hypotheses. Any KS can generate or modify a hypothesis. The blackboard represents intermediate states of problem-solving activity (candidate facts) and communicates messages from one KS that activates other KSs (beliefs).

Life-long learning: growing area

Progression

•Banko and Etzioni discuss the growing field of information extraction from the abundance of electronic Traditional IE systems focused on locating instances of **narrow pre-specified relations**, such as time and place of events, from **small homogeneous corpora**.

• Etzioni et. Al, 2005 Unsupervised named-entity extraction from the web: An experimental study.Artificial Intelligence, the KnowItAll system scaled this to the size and diversity of relationships present in the **millions of web pages**, **but required a user to name a relation** prior to each extraction cycle.

•Open IE attempts to automatically discover all possible relations. Shinyama and Sekine 2006 Preemptive information extraction using unrestricted relation discovery was applied to a **small corpus of newswire articles**.

•Banko and Etzioni, 2007 first introduced the TextRunner system that processed over 110,000,000 web pages and yielded over 330,000,000 **statements about concrete entities** with a precision of 88%.

•Banko and Etzioni introduce Alice which automatically builds a collection of **concepts, facts and generalizations about a particular topic**, directly from Web text with a precision of 78%.

•Compared to Carlson et al, NELL expanded this to any domain, with a precision of 74%

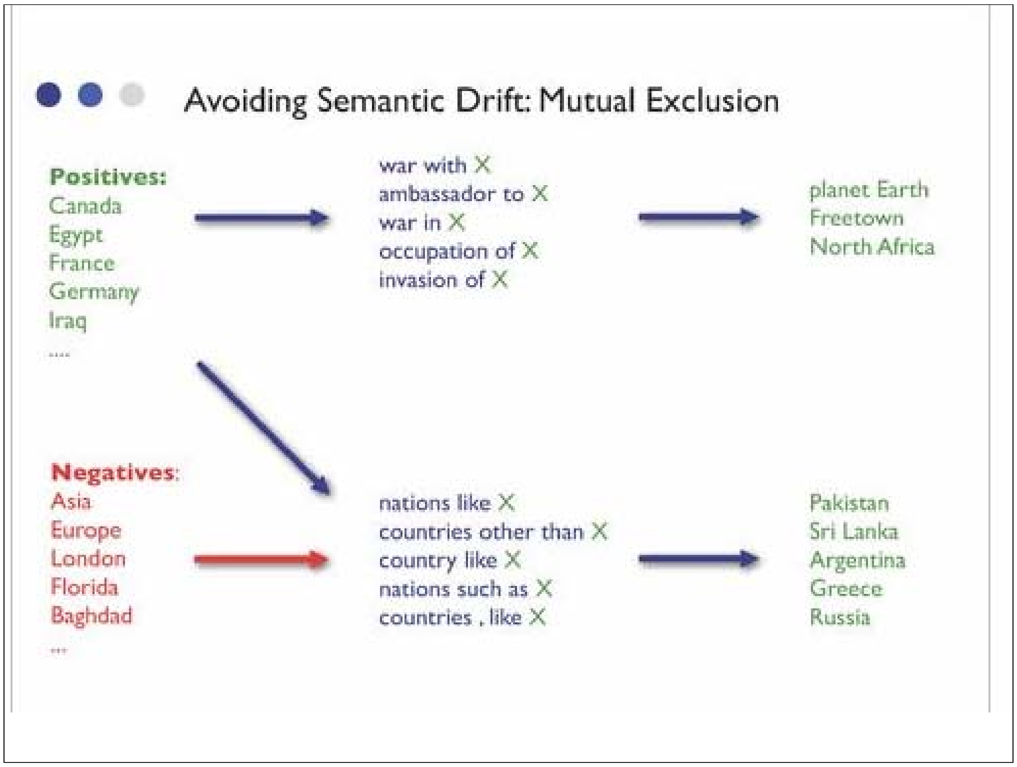
-Thrun and Mitchell compare lifelong learning to humans who use learning from a first task to simplify learning the second, and applying this learning across environments. As an example they point to Singh and Lin 1992, who demonstrated that a robot was able to solve more complex navigation tasks when they were given simpler related learning tasks beforehand, but unable to learn the same complex tasks in isolation.

-Thrun and Mitchell describe the concept of life-long learning in terms of bootstrapping learning algorithms that transfer learned knowledge from one learning task to another.

Semi-supervised boostrap learning: (bootstrap – sample with replacement or reuse)

-Semi-supervised bootstrap learning methods begin with a small set of labeled data, train a model, then use that model to label more data. This approach is used in many applications. For example, Yarowsky used bootstrap learning in an unsupervised system to train classifiers for word sense disambiguation by identifying a small number of examples of each word sense of a given word, and training the system on those examples. Blum and Mitchell use bootstrapping cotraining for web page classification by using a small set of examples to identify weak predictors on web pages and on links. They then use these combined weak indicators to further train the learning algorithm.

-Used alone can result in semantic drift



Avoiding Semantic Drift: Mutual Exclusion

Positives:
Canada
Egypt
France
Germany
Iraq
....

war with X
ambassador to X
war in X
occupation of X
invasion of X

planet Earth
Freetown
North Africa

Negatives:
Asia
Europe
London
Florida
Baghdad
...

nations like X
countries other than X
country like X
nations such as X
countries , like X

Pakistan
Sri Lanka
Argentina
Greece
Russia

Taken from slides presented on videolectures.net titled – Coupled Semi-Supervised Learning for Information Extraction given by Andrew Carlson.

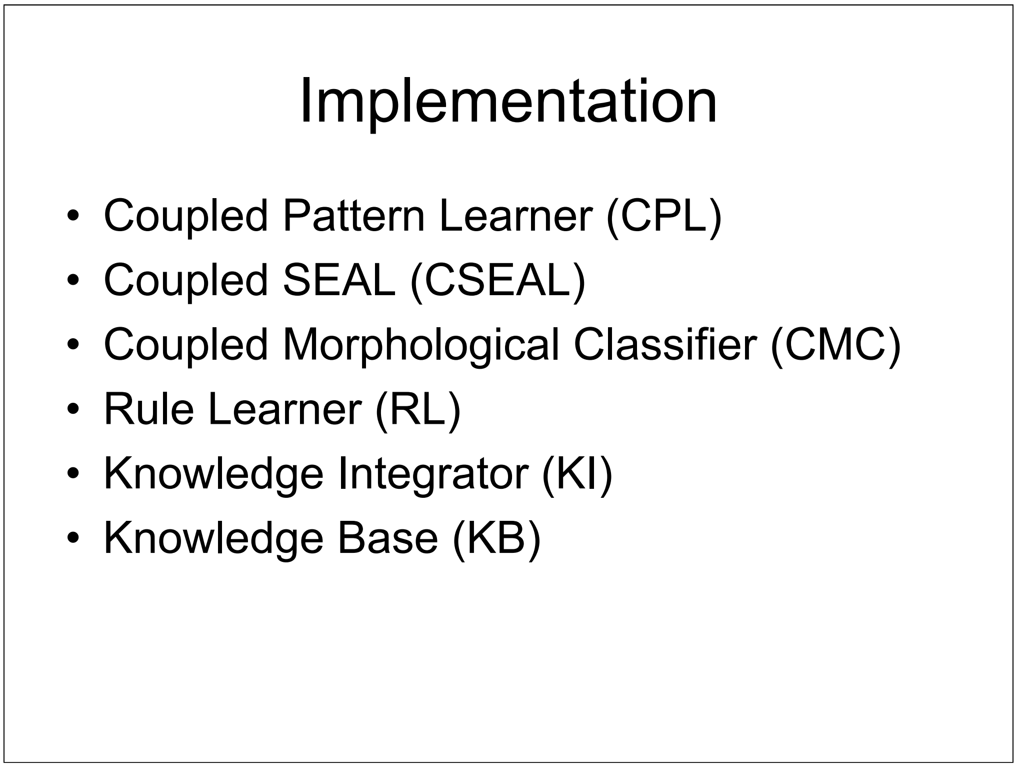
Semantic drift – come up with entities that are not countries.

Coupled semi-supervised learning: In an earlier paper, Coupled Semi-Supervised Learning for Information Extraction, Carlson et al, introduce the idea of coupling training by adding constraints, and show a significant improved of the coupled extractors over the uncoupled ones. Types of coupling used are:

-mutual exclusion – mutually exclusive predicates cannot both be satisfied by the same input x, show example - use known negative examples to eliminate relations.

-Relation argument type checking – ie arguments of CompanyIsInEcomnomicSector relation are of category type Company and EconomicSector

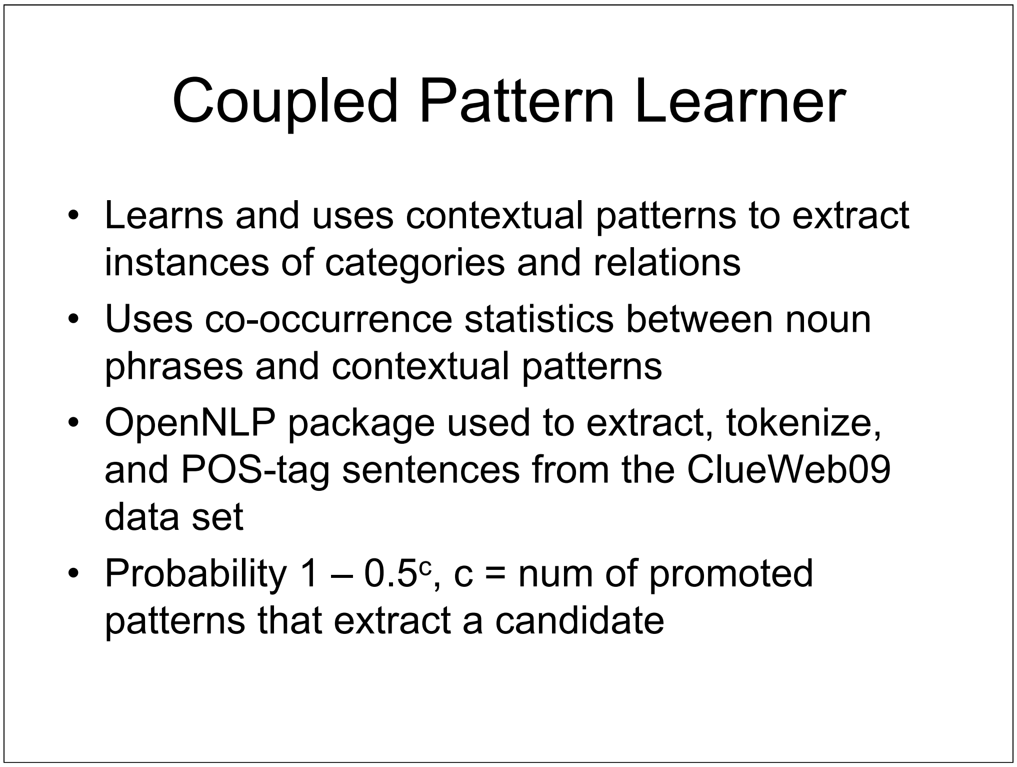
-Unstructured and semi-structured text feature – train on free-form text and html wrappers, require that they both provide the same classification label



Implementation

- Coupled Pattern Learner (CPL)
- Coupled SEAL (CSEAL)
- Coupled Morphological Classifier (CMC)
- Rule Learner (RL)
- Knowledge Integrator (KI)
- Knowledge Base (KB)

Components in NELL



Coupled Pattern Learner

- Learns and uses contextual patterns to extract instances of categories and relations
- Uses co-occurrence statistics between noun phrases and contextual patterns
- OpenNLP package used to extract, tokenize, and POS-tag sentences from the ClueWeb09 data set
- Probability 1 – 0.5^c, c = num of promoted patterns that extract a candidate

•Free text extractor

•Example of contextual pattern “mayor of X”, X plays for Y”

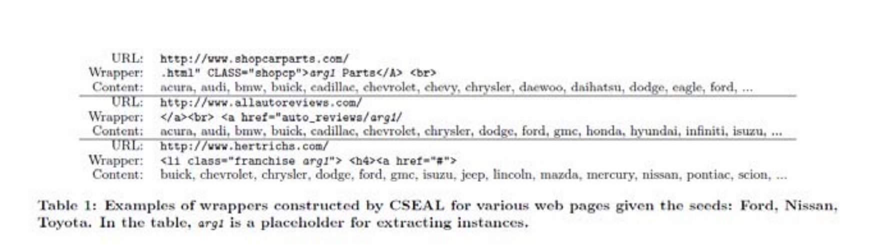
•Nouns and patterns defined by POS tag sequences

•2 billion sentences generated from 500 million web page English portion of the ClueWeb09 data set – Carnegie Mellon University– used a web crawler to generate 1 billion pages in 10 languages.

•stances are assigned a probability

Coupled SEAL

- Queries the Internet and extracts novel instances
- Uses mutual exclusion to filter results
- 5 queries/category and 10 queries/relation
- fetches 50 web pages/query
- Probability 1 – 0.5^c, c = num of unfiltered wrappers that extract an instance



URL:	`http://www.shopcarparts.com/`
Wrapper:	`.html" CLASS="shopcp">arg1 Parts</A>  `
Content:	acura, audi, bmw, buick, cadillac, chevrolet, chevy, chrysler, daewoo, daihatsu, dodge, eagle, ford, ...
URL:	`http://www.allautoreviews.com/`
Wrapper:	`</a>  <a href="auto_reviews/arg1/`
Content:	acura, audi, bmw, buick, cadillac, chevrolet, chrysler, dodge, ford, gmc, honda, hyundai, infiniti, isuzu, ...
URL:	`http://www.hertrichs.com/`
Wrapper:	`<li class="franchise arg1"> <h4><a href="#">`
Content:	buick, chevrolet, chrysler, dodge, ford, gmc, isuzu, jeep, lincoln, mazda, mercury, nissan, pontiac, scion, ...

Table 1: Examples of wrappers constructed by CSEAL for various web pages given the seeds: Ford, Nissan, Toyota. In the table, *arg1* is a placeholder for extracting instances.

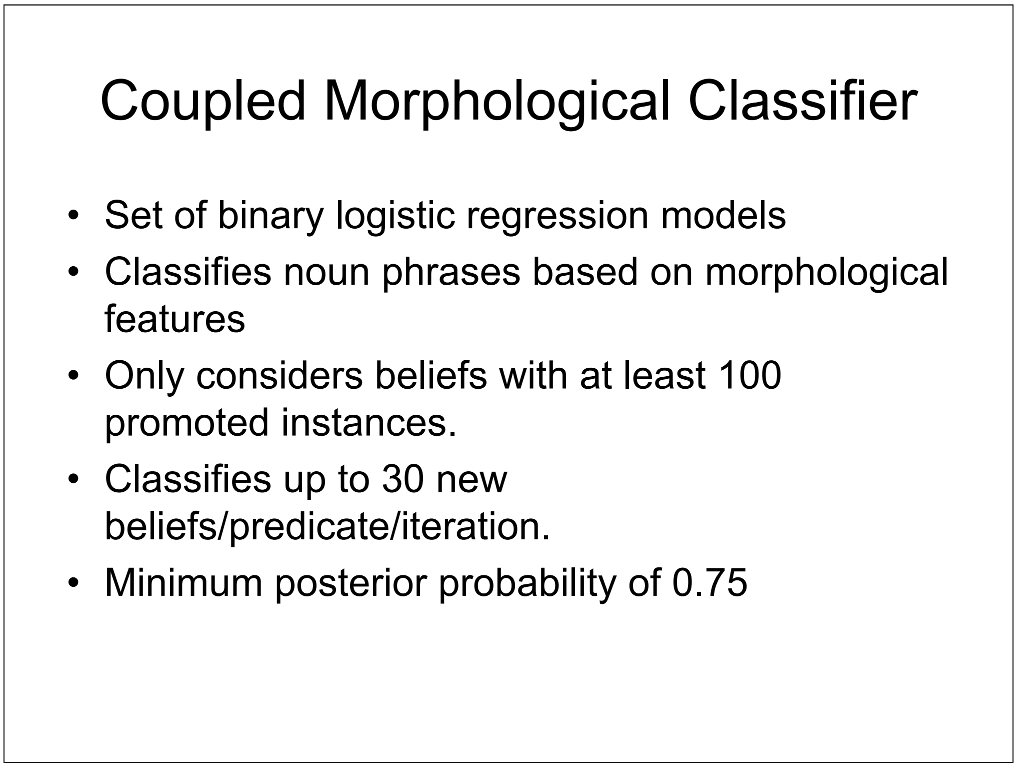
•Semi-structured extractor

•Queries the internet with sets of beliefs from each category or relation, and mines results to extract novel instances of the corresponding predicate.

•Uses mutual exclusion relationships to provide negative examples which are used to filter out overly general lists and tables.

•Sub-samples the set of seed instances, and is configured to issue 5 queries per category and 10 queries per relation.

•Candidate facts are assigned a probability



Coupled Morphological Classifier

- Set of binary logistic regression models
- Classifies noun phrases based on morphological features
- Only considers beliefs with at least 100 promoted instances.
- Classifies up to 30 new beliefs/predicate/iteration.
- Minimum posterior probability of 0.75

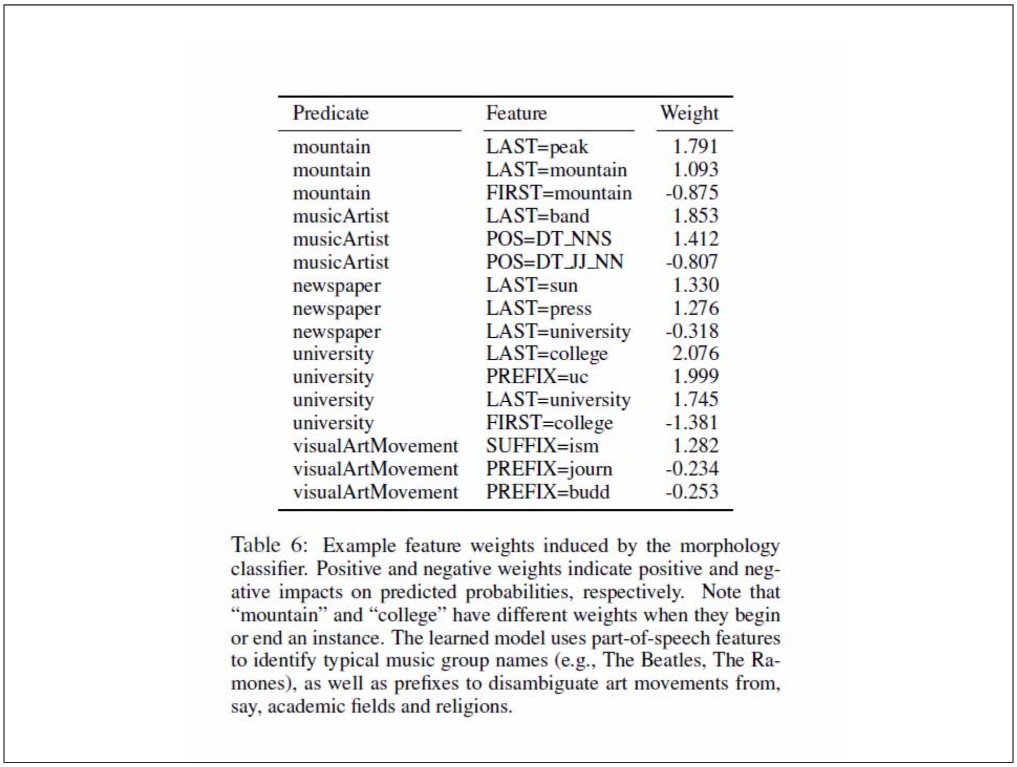
•Consists of a set of L2-regularized logistic regression models – one per category

•**A logistic regression** model is used to predict the probability of occurrence of an event by fitting data to a logistic curve. Like many forms of regression analysis, it makes use of several predictor variables that may be either numerical or categorical. For example, the probability that a person has a heart attack within a specified time period might be predicted from knowledge of the person's age, sex and body mass index.

•The sample complexity of L2-regularized logistic regression is linear in the number of features as compared to logarithmic for L1

•Classifies nouns based on features such as words, capitalization, affixes, POS, etc.

•Posterior = The probability after the evidence has been examined.

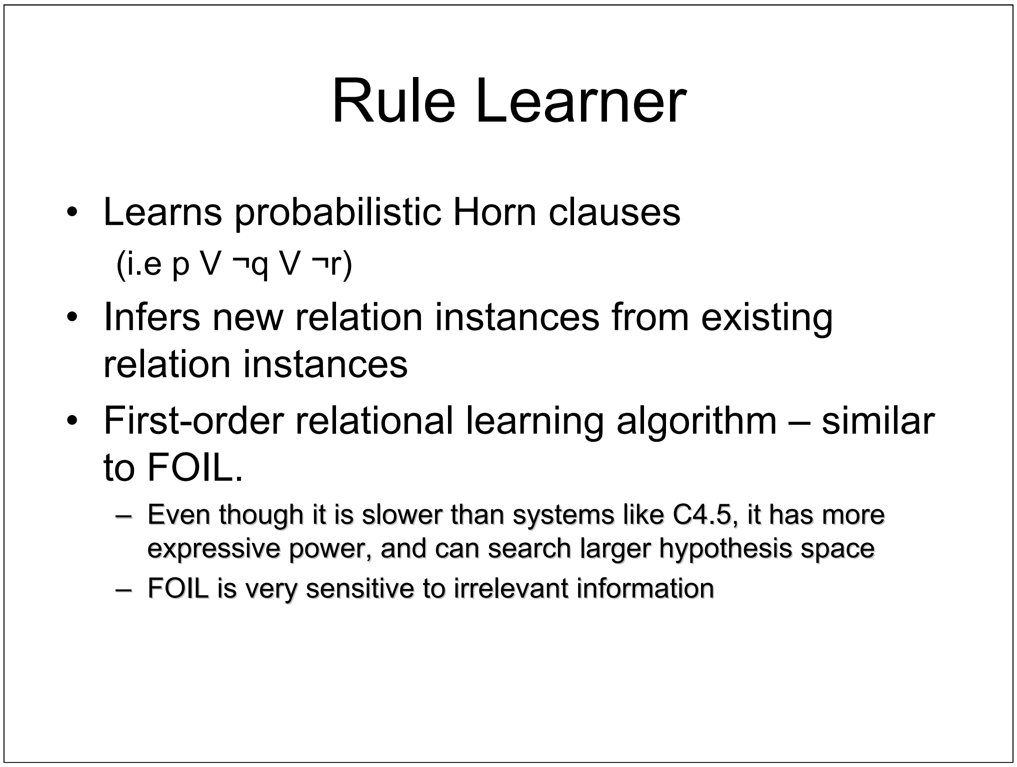


Predicate	Feature	Weight
mountain	LAST=peak	1.791
mountain	LAST=mountain	1.093
mountain	FIRST=mountain	-0.875
musicArtist	LAST=band	1.853
musicArtist	POS=DT_NNS	1.412
musicArtist	POS=DT_JJ_NN	-0.807
newspaper	LAST=sun	1.330
newspaper	LAST=press	1.276
newspaper	LAST=university	-0.318
university	LAST=college	2.076
university	PREFIX=uc	1.999
university	LAST=university	1.745
university	FIRST=college	-1.381
visualArtMovement	SUFFIX=ism	1.282
visualArtMovement	PREFIX=journ	-0.234
visualArtMovement	PREFIX=budd	-0.253

Table 6: Example feature weights induced by the morphology classifier. Positive and negative weights indicate positive and negative impacts on predicted probabilities, respectively. Note that "mountain" and "college" have different weights when they begin or end an instance. The learned model uses part-of-speech features to identify typical music group names (e.g., The Beatles, The Ramones), as well as prefixes to disambiguate art movements from, say, academic fields and religions.

-A phrase that ends in "peak" has a high probability of being the name of a mountain.

-But a phrase that begins with "mountain" has a low probability of being the name of a mountain.



Rule Learner

- Learns probabilistic Horn clauses
  (i.e p V ¬q V ¬r)
- Infers new relation instances from existing relation instances
- First-order relational learning algorithm – similar to FOIL.
  - Even though it is slower than systems like C4.5, it has more expressive power, and can search larger hypothesis space
  - FOIL is very sensitive to irrelevant information

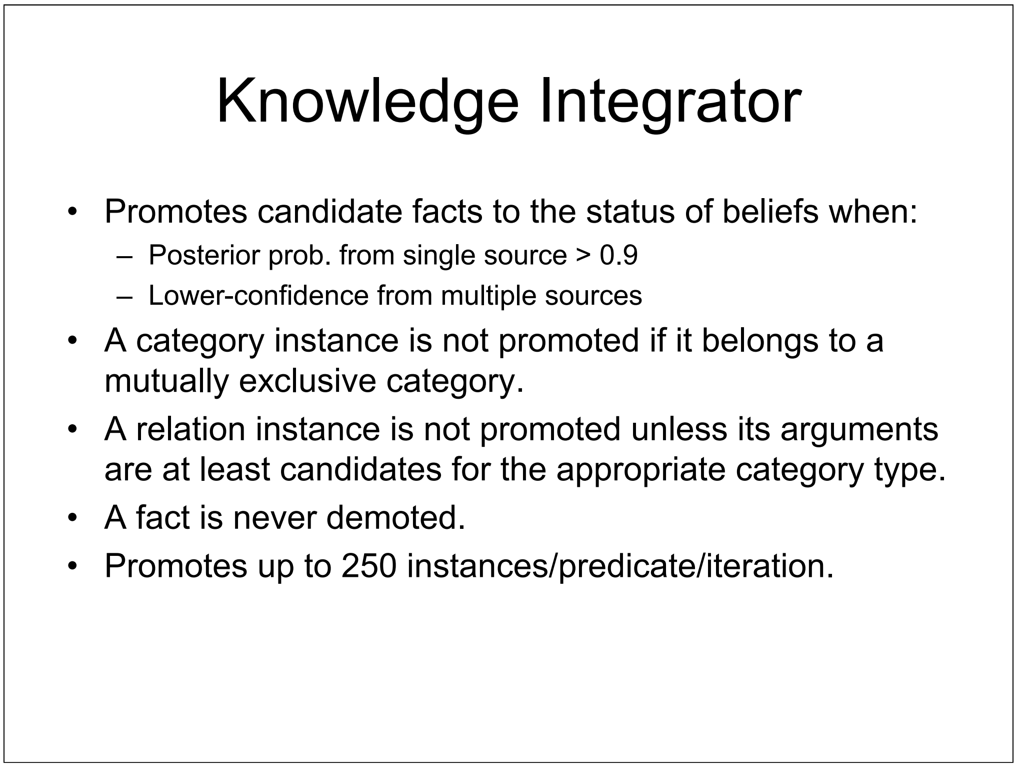
•Horn clauses – a clause with at most 1 positive literal

•The new learned rules are used to infer new relation instances from other relation instances that are already in the KB.

•Learning algorithm similar to FOIL Quinlan and Cameron-Jones 1993)

FOIL – First Order Learning System

- Outer loop of the algorithm:
  - Divide training data into positive instances and negative instances
  - Find a rule that covers some of the positive instances (+ tuples)
  - Store the rule and remove the covered positive data from the dataset
  - Re-run the algorithm on the reduced dataset until no positive instances are left
- Many requirements are still unsolved
  - Construction of new predicates, Strategy for constructing programs etc…



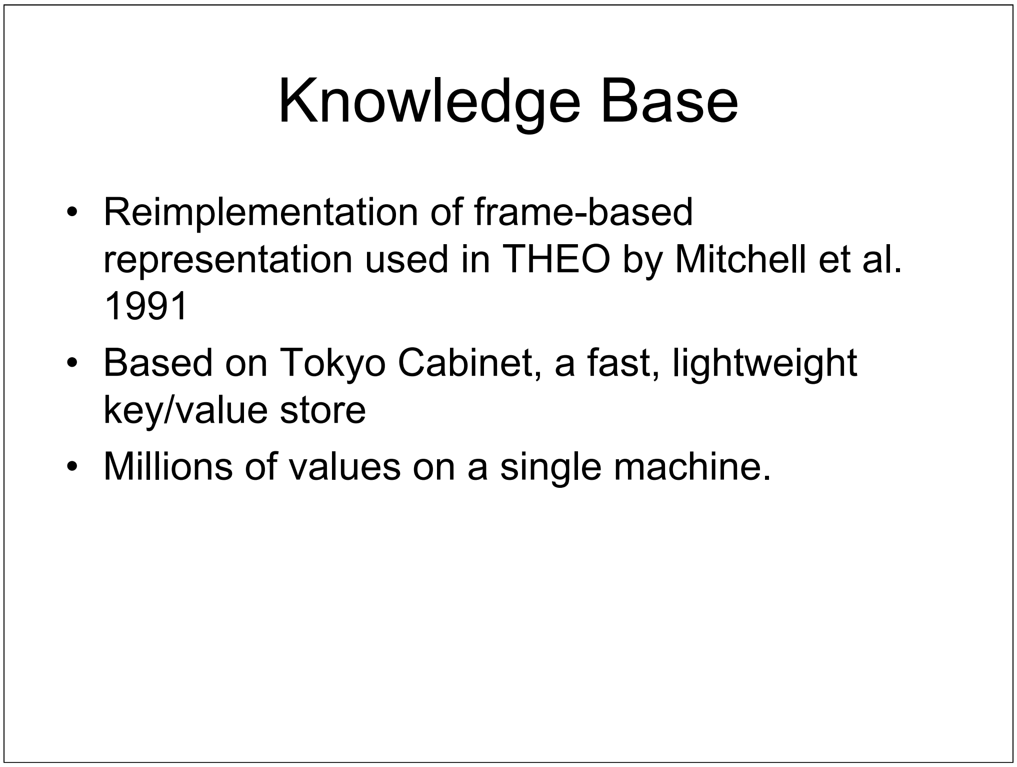
Knowledge Integrator

- Promotes candidate facts to the status of beliefs when:
  - Posterior prob. from single source > 0.9
  - Lower-confidence from multiple sources
- A category instance is not promoted if it belongs to a mutually exclusive category.
- A relation instance is not promoted unless its arguments are at least candidates for the appropriate category type.
- A fact is never demoted.
- Promotes up to 250 instances/predicate/iteration.

•No minimum probability is provided for the lower-confidence. Should require a min prob. from a min number of sources.

•Relation - .. And are not already believed to be instances of a category that is mutually exclusive.

•Configured to 250 instance max, but threshold was rarely hit during the experiments.



Knowledge Base

- Reimplementation of frame-based representation used in THEO by Mitchell et al. 1991
- Based on Tokyo Cabinet, a fast, lightweight key/value store
- Millions of values on a single machine.

- THEO, a Frame-based representation – a software framework to support development of self-modifying problem solving systems.
  - Beliefs are represented as values of slots of frames
    - (<entity><slot>) = <value>, ie. (fred wife) = wilma
  - Problems are represented by slot instances whose values are not yet known
    - (<entity><slot>), ie. (fred wife) is a problem whose solution is wilma
    - One to one correspondence between beliefs and problems, a slot is the name of a relation, a belief is an instance of that relation, and a problem is a query that specifies a relation name, and an element of its domain, and asks for the corresponding element of the relation's range ie, the answer.

-Based on Tokyo Cabinet – library of routines to manage a database. The database is a simple data file containing records, each is a pair of a key and a value. Every key and value is serial bytes with variable length. Both binary data and character string can be used as a key and a value. There is neither concept of data tables nor data types. Records are organized in hash table, B+ tree, or fixed-length array.

-can handle millions of values on a single machine.

Tokyo Cabinet is developed as the successor of GDBM and QDBM on the following purposes. They are achieved and Tokyo Cabinet replaces conventional DBM products.

improves space efficiency : smaller size of database file.

improves time efficiency : faster processing speed.

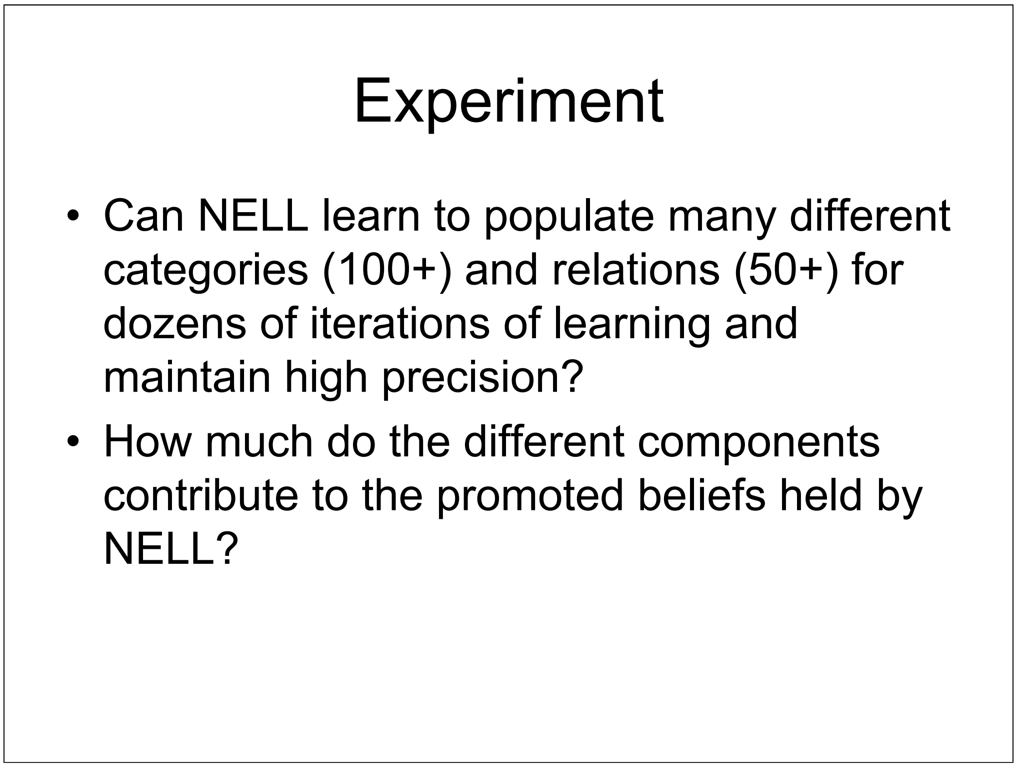
improves parallelism : higher performance in multi-thread environment.

improves usability : simplified API.

improves robustness : database file is not corrupted even under catastrophic situation.

supports 64-bit architecture : enormous memory space and database file are available.

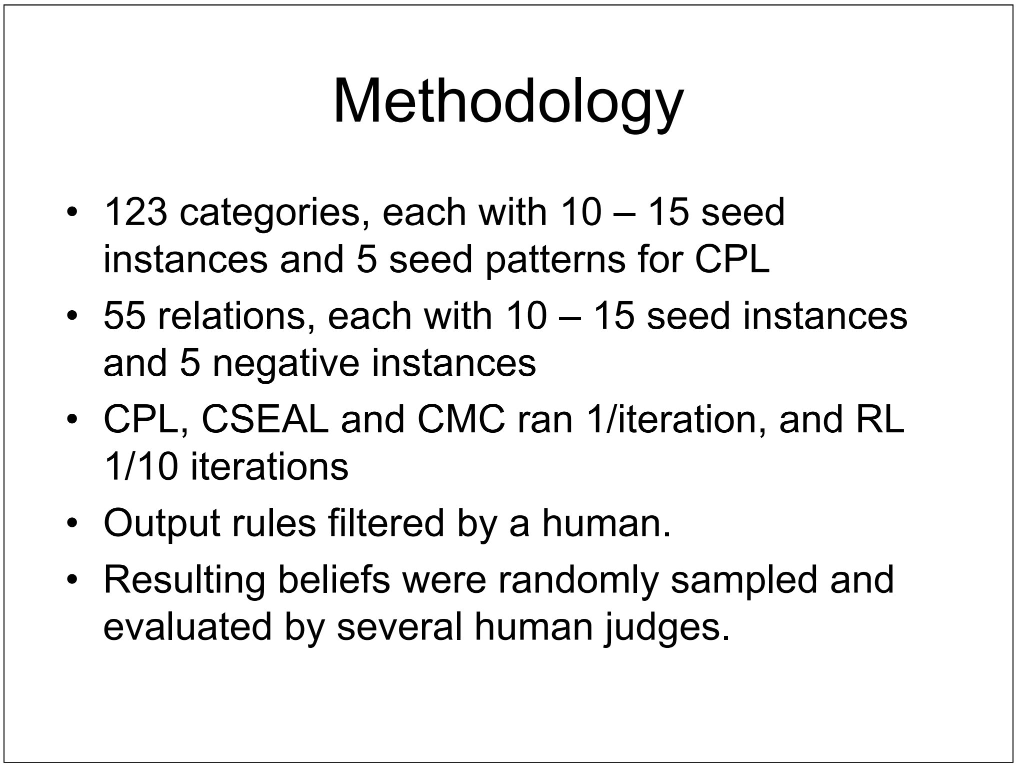
Tokyo Cabinet is written in the C language, and provided as API of C, Perl, Ruby, Java, and Lua. Tokyo Cabinet is available on platforms which have API conforming to C99 and POSIX. Tokyo Cabinet is a free software licensed under the GNU Lesser General Public License.



Experiment

- Can NELL learn to populate many different categories (100+) and relations (50+) for dozens of iterations of learning and maintain high precision?
- How much do the different components contribute to the promoted beliefs held by NELL?

Explore the following questions.



Methodology

- 123 categories, each with 10 – 15 seed instances and 5 seed patterns for CPL
- 55 relations, each with 10 – 15 seed instances and 5 negative instances
- CPL, CSEAL and CMC ran 1/iteration, and RL 1/10 iterations
- Output rules filtered by a human.
- Resulting beliefs were randomly sampled and evaluated by several human judges.

•Seed patterns derived from Hearst 1992 – described a method of automatic acquisitions of the hyponymy lexical relation (is-a) from unrestricted text, that avoided the need for pre-encoded knowledge. They identified a set of lexico-syntactic patterns that are easily recognizable, that occur frequently and across text genres, and that indisputably indicate the lexical relation of interest.

NP as {NP ,} {(or [and)} NP*

... works by such authors as Herrick, Goldsmith, and Shakespeare.

➔ *hyponym (“author", “Herrick'), Hyponym("author", "(Goldsmith "), hyponynl("author", "Shakespeare")*

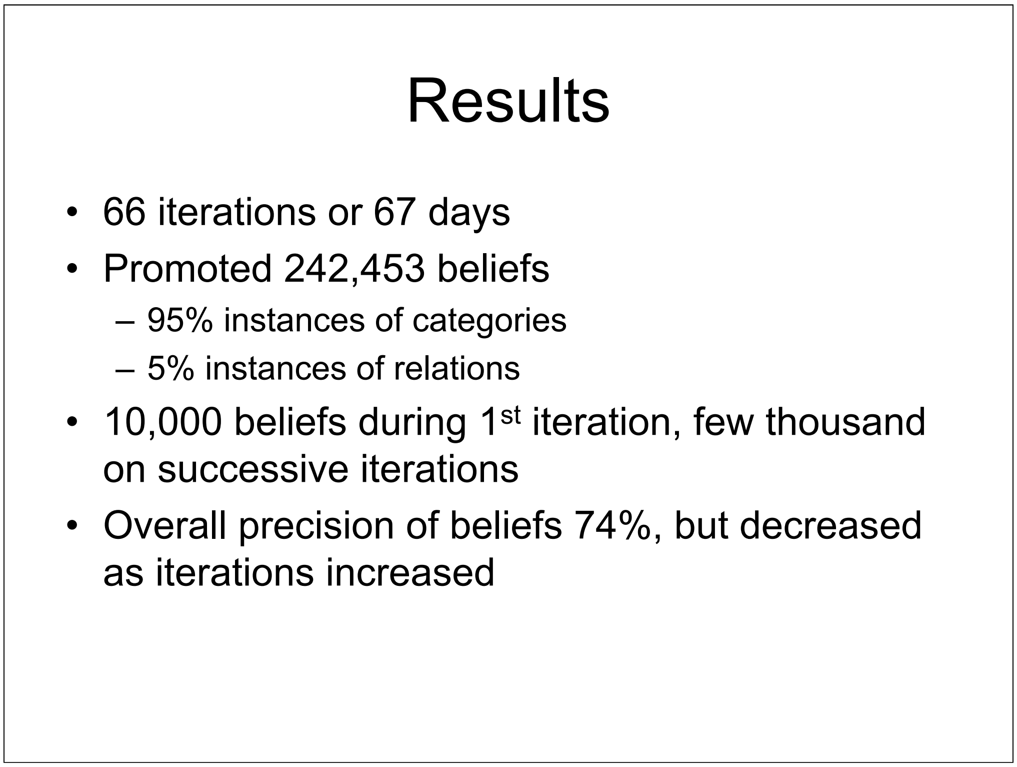
•Categories included locations (mountains, museums), people (scientists, writers), animals (reptiles, birds), organizations (companies, web sites, sports teams), and others.

•Relations captured relationships between different categories (e.g. teamPlaysSport, bookWriter, companyProducesProduct).

•Manual approval of rules took only a few minutes. – what type of rules were filtered out?

•Cases of disagreement were discussed in detail before a decision was made.

•Temporal scope was ignored, so facts which were no longer true, were considered correct, i.e. a former coach of a sports team.

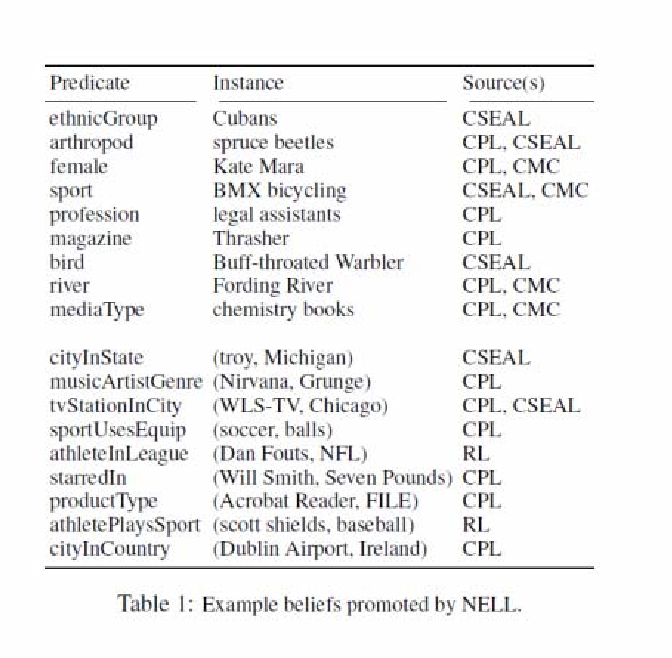


Results

- 66 iterations or 67 days
- Promoted 242,453 beliefs
  - 95% instances of categories
  - 5% instances of relations
- 10,000 beliefs during 1st iteration, few thousand on successive iterations
- Overall precision of beliefs 74%, but decreased as iterations increased

•Continued to promote beliefs, so the potential exists to learn more.

•Number of iterations 1 – 22 : 90%, 23 – 44 : 71%, 45 – 66 : 57%



Predicate	Instance	Source(s)
ethnicGroup	Cubans	CSEAL
arthropod	spruce beetles	CPL, CSEAL
female	Kate Mara	CPL, CMC
sport	BMX bicycling	CSEAL, CMC
profession	legal assistants	CPL
magazine	Thrasher	CPL
bird	Buff-throated Warbler	CSEAL
river	Fording River	CPL, CMC
mediaType	chemistry books	CPL, CMC
cityInState	(troy, Michigan)	CSEAL
musicArtistGenre	(Nirvana, Grunge)	CPL
tvStationInCity	(WLS-TV, Chicago)	CPL, CSEAL
sportUsesEquip	(soccer, balls)	CPL
athleteInLeague	(Dan Fouts, NFL)	RL
starredIn	(Will Smith, Seven Pounds)	CPL
productType	(Acrobat Reader, FILE)	CPL
athletePlaysSport	(scott shields, baseball)	RL
cityInCountry	(Dublin Airport, Ireland)	CPL

Table 1: Example beliefs promoted by NELL.

- Beliefs and the component that found them.

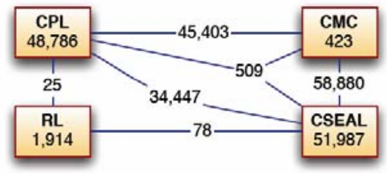
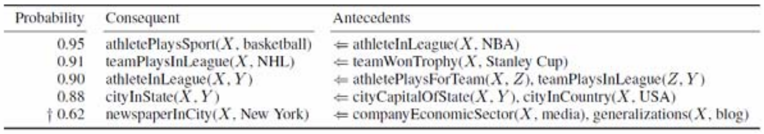


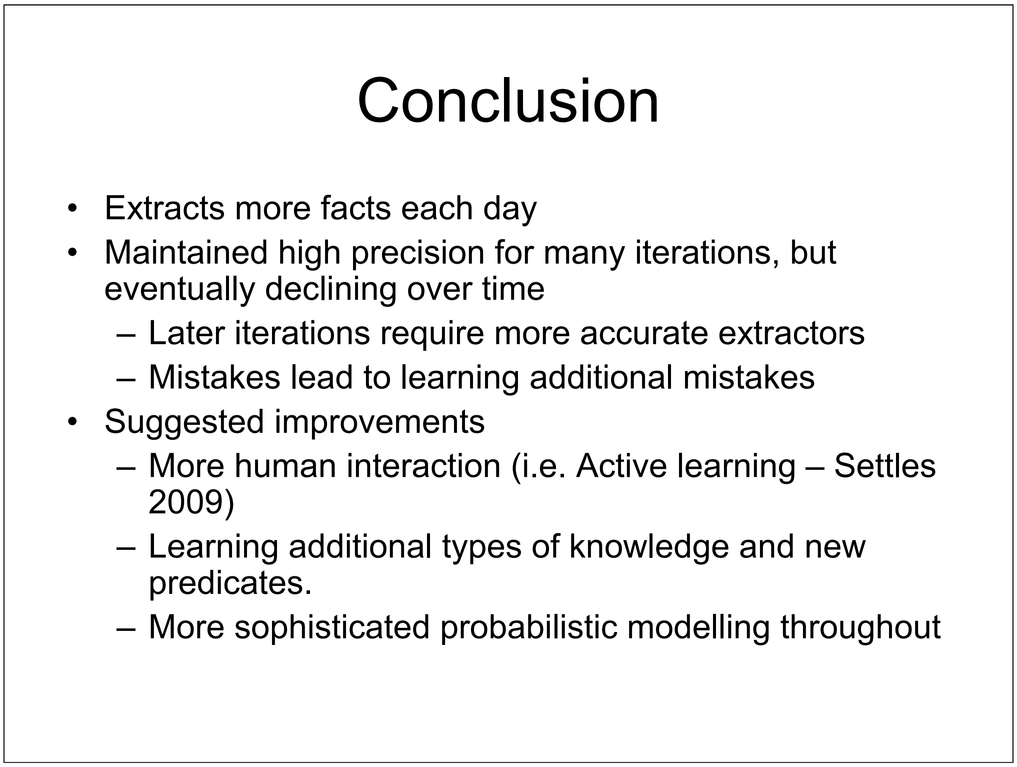
Figure 3: Source counts for beliefs promoted by NELL after 66 iterations. Numbers inside nodes indicate the number of beliefs promoted based solely on that component. Numbers on edges indicate beliefs promoted based on evidence from multiple components.



Probability	Consequent	Antecedents
0.95	athletePlaysSport(X, basketball)	⇐ athleteInLeague(X, NBA)
0.91	teamPlaysInLeague(X, NHL)	⇐ teamWonTrophy(X, Stanley Cup)
0.90	athleteInLeague(X, Y)	⇐ athletePlaysForTeam(X, Z), teamPlaysInLeague(Z, Y)
0.88	cityInState(X, Y)	⇐ cityCapitalOfState(X, Y), cityInCountry(X, USA)
† 0.62	newspaperInCity(X, New York)	⇐ companyEconomicSector(X, media), generalizations(X, blog)

Table 7: Example horn clauses induced by the rule learner. Probabilities indicate the conditional probability that the literal to the left of ⇐ is true given that the literals to the right are satisfied. Each rule captures an empirical regularity among the relations mentioned by the rule. The rule marked with † was rejected during human inspection.

- Inside box, number of beliefs promoted due to a specific component.
- Lines indicate number of beliefs promoted as a result of multiple components.
- CPL and CSEAL for many beliefs, but more than half of the beliefs were based on multiple sources.
- Beliefs promoted by RL are fairly independent from the other components. RL learned an average of 66.5 novel rules per iteration, of which 92% were approved.



Conclusion

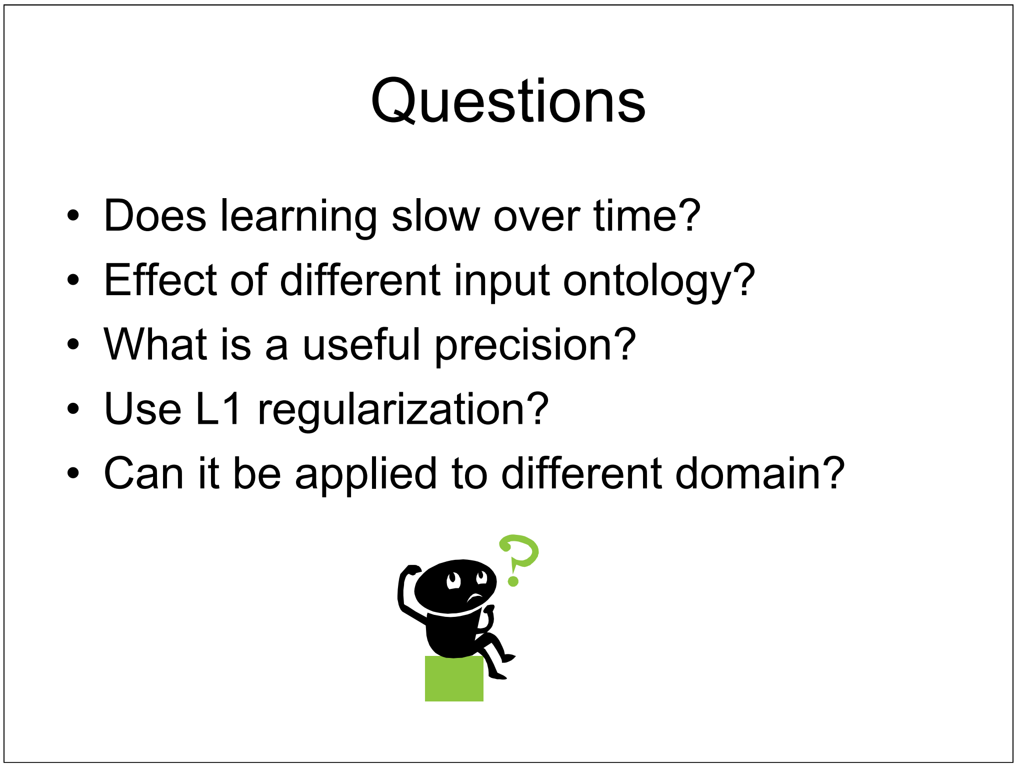
- Extracts more facts each day
- Maintained high precision for many iterations, but eventually declining over time
  - Later iterations require more accurate extractors
  - Mistakes lead to learning additional mistakes
- Suggested improvements
  - More human interaction (i.e. Active learning – Settles 2009)
  - Learning additional types of knowledge and new predicates.
  - More sophisticated probabilistic modelling throughout

•Learning to learn.

•Easier extractions occur during early iterations.

•CPL and CMC not perfectly uncorrelated in their errors i.e. for the category bakedGood, CPL learns the pattern “X are enable in” because of the believed instance “cookies”, which causes CPS to extract “persistent cookies” as a candidate for bakedGood.

•Active learning – allow NELL to ask “queries” about its beliefs, theories or features – for example, “X are enabled in” is likely to be rare in the bakedGood category. If NELL could identify the issue, it could develop a query for the human interaction that could correct the error.



Questions

- Does learning slow over time?
- Effect of different input ontology?
- What is a useful precision?
- Use L1 regularization?
- Can it be applied to different domain?

•Within the limitation of the experiment, learning didn’t slow down, but over time would you have to search farther and have problems comparing to the existing KB as it grows over time?

•Would the input ontology produce different results?

•After 45 days, precision fell to 57%

•Feature selection, L1 vs.L2 regularization, and rotational invariance Andrew Ng ICML 2004 shows that L2 regularization classifies poorly for even a few irrelevant features.

•Can it be applied to different domain i.e cell biology?