Never Ending Learning

Tom M. Mitchell

Justin Betteridge, Andy Carlson, Jamie Callan, William Cohen, Estevam Hruschka, Bryan Kisiel, Mahaveer Jain, Jayant Krishnamurthy, Edith Law, Thahir Mohamed, Burr Settles, Richard Wang

Machine Learning Department
Carnegie Mellon University

June 2010
Humans learn many things, for years, and become better learners over time.

Why not machines?
Never Ending Learning

Task: acquire a growing competence without asymptote
• over years
• multiple functions
• where learning one thing improves ability to learn the next
• acquiring data from humans, environment

Many candidate domains:
• Robots
• Softbots
• Game players
Years of Relevant ML Research

• Architectures for problem solving/learning
  – SOAR [Newell, Laird, Rosenbloom 1986]
  – ICARUS [Langley], PRODIGY [Carbonell], …

• Life long learning, transfer learning, multi-label learning
  – EBNN [Thrun & Mitchell 1993]
  – Learning to learn [Thrun & Pratt, 1998]

• Transfer learning
  – Multitask learning [Caruana 1995]
  – Transfer reinforcement learning [Parr & Russell 1998]
  – Multilabel data [ICML 2010], Learning with structured outputs

• Active Learning
  – see survey: [Settles 2010];
  – Multi-task active learning [Harpale & Yang, ICML 2010]

• Curriculum learning
NELL: Never-Ending Language Learner

Inputs:
• initial ontology
• handful of examples of each predicate in ontology
• the web
• occasional interaction with human trainers

The task:
• run 24x7, forever
• each day:
  1. extract more facts from the web to populate the initial ontology
  2. learn to read (perform #1) better than yesterday
Why Do This?

1. Case study in Never-Ending Learning

2. New approach to natural language understanding
   - *Micro-reading*: sentence $\rightarrow$ content
   - *Macro-reading*: corpus, ontology $\rightarrow$ populated ontology

3. Build the world’s largest structured KB
   - AI is right: intelligence requires knowledge
NELL: Never-Ending Language Learner

Goal:
- run 24x7, forever
- each day:
  1. extract more facts from the web to populate given ontology
  2. learn to read better than yesterday

Today…

Running 24 x 7, since January, 2010

Input:
- ontology defining ~200 categories and relations
- dozen seed examples of each
- 500 million web pages (ClueWeb – Jamie Callan)

Result:
- continuously growing KB with ~300,000 extracted beliefs
ibm:

generalizations = {company}
    candidateValues = {conference, company, product}
headquarteredIn = armonk
    candidateValues = {armonk}
producesProduct = {pc}
    candidateValues = {domino, thinkpad_line, ibm_e_business_logo, first_pcs, powerpc, internet, ibm_pc, iseries, rational, first_pc, quickplace, first_ibm_pc, vga_controller, original_pc, at_computer, wsfl_specification, selectric, pc, pc_convertible, workplace_client_technology, workplace, ids, opteron_server, linux_strategy, very_interesting_study, video_graphics_array, business_partner_emblem, ibm, …}
acquired = {iss, cognos, informix}
    candidateValues = {spi, watchfire, telelogic, daksh, lotus, iss, internet_security_systems, gluecode, cognos, sequent, tivoli, diligent, informix, webify_solutions, geronimo, rational, information_laboratory, meiosys, webify, …}
acquiredBy = lenovo_group
    candidateValues = {lenovo_group, lenovo, china, arsenal}
competesWith = {sun, texas_instruments, samsung, hewlett_packard, apple, novell, oracle, microsoft, ricoh, hp, amazon}
companyEconomicSector = {software}
hasOfficeInCountry = {united_states, canada, usa, germany, england, uk, france}
    candidateValues = {san_jose, dallas, cambridge, europe, boca_raton, boulder, united_states, tucson, november, new_york, poughkeepsie, canada, october, united, research_triangle_park, rochester, beaverton, armonk, usa, u_s, germany, new_delhi, boeblingen, england, uk, france, us, facebook, masters_degree}
Semi-Supervised Bootstrap Learning

Extract cities:

Paris  Pittsburgh  Seattle  Cupertino

San Francisco  Austin  denial

anxiety  selfishness  Berlin

mayor of arg1  live in arg1

arg1 is home of traits such as arg1

it's underconstrained!!
Key Idea 1: Coupled semi-supervised training of many functions

hard (underconstrained) semi-supervised learning problem

much easier (more constrained) semi-supervised learning problem
person

NP
Coupled Training Type 1: Co-Training, Multiview, Co-regularization

\[ x = \langle x_1, x_2 \rangle \]

Constraint: \( f_1(x_1) = f_2(x_2) \)

References:
- Blum & Mitchell; 98
- Dasgupta et al; 01
- Ganchev et al., 08
- Sridharan & Kakade, 08
- Wang & Zhou, ICML10
Coupled Training Type 1: Co-Training, Multiview, Co-regularization

Constraint: \( f_1(x_1) = f_2(x_2) \)

If \( f_1, f_2 \) PAC learnable, \( X_1, X_2 \) conditionally indep
Then PAC learnable from unlabeled data and weak initial learner

and disagreement between \( f_1, f_2 \) bounds error of each
Type 1 Coupling Constraints in NELL

NP:

- NP text context distribution
- NP morphology
- NP HTML contexts

- ___ is a friend
- ___ rang the ___
- ___ walked in
- capitalized?
- ends with ‘...ski’?
- contains “univ.”?
Coupled training type 2
Structured Outputs, Multitask,
Posterior Regularization, Multilabel

Learn functions with same input, different outputs, where we know some constraint $\Phi(Y_1, Y_2)$

Constraint: $\Phi(f_1(x), f_2(x))$

Effectiveness $\sim$ probability that $\Phi(Y_1, Y_2)$ will be violated by incorrect $f_j$ and $f_k$

[Daume, 2008]
[Bakhir et al., eds. 2007]
[Roth et al., 2008]
[Taskar et al., 2009]
[Carlson et al., 2009]
Type 2 Coupling Constraints in NELL

![Diagram showing relationships between athlete, coach, person, sport, and team.]
Multi-view, Multi-Task Coupling

C categories, V views, CV = 170*3=510 coupled functions
	pairwise constraints on functions ≈ 10^5
Learning Relations between NP’s
Type 3 Coupling: Argument Types

Constraint: $f_3(x_1, x_2) \rightarrow (f_1(x_1) \text{ AND } f_2(x_2))$
NELL: 550+ fns, coupled via $10^5$ constraints

**Functions**

NP Morphology $\rightarrow$ fruit
NP Text Context $\rightarrow$ fruit
NP HTML Context $\rightarrow$ fruit
...
NP Morphology $\rightarrow$ city
NP Text Context $\rightarrow$ city
NP HTML Context $\rightarrow$ city
...
NP1, NP2 $\rightarrow$ mayorOf
TextContext $\rightarrow$ mayorOf
HTMLcontext $\rightarrow$ mayorOf

**Constraint Vocabulary**

agree($\text{fruit}(\text{NP\_Morphology}),$
          $\text{fruit}(\text{NP\_TextContext}))$
...
mutuallyExclusive($\text{fruit}, \text{city}$)
...
subset($\text{city}, \text{location}$)
...
argumentTypes($\text{mayorOf},$
                 $\text{city}, \text{politician}$)
...
Pure EM Approach to Coupled Training

**E**: jointly estimate latent labels for each function of each unlabeled example

**M**: retrain all functions, based on these probabilistic labels

Scaling problem:
- **E** step: 20M NP’s, $10^{14}$ NP pairs to label
- **M** step: 50M text contexts to consider for each function → $10^{10}$ parameters to retrain
- even more URL-HTML contexts…
NELL’s Approximation to EM

E’ step:
• Consider only a growing subset of the latent variable assignments
  – category variables: up to 250 NP’s per category per iteration
  – relation variables: add only if confident and args of correct type
  – this set of explicit latent assignments IS the knowledge base

M’ step:
• Each view-based learner retrains itself from the updated KB
• “context” methods create growing subsets of contexts
NELL’s Approximation to EM

E’ step:
• Consider only a growing subset of the latent variable assignments
  – category variables: up to 250 NP’s per category per iteration
  – relation variables: add only if confident and args of correct type
  – this set of explicit latent assignments is the knowledge base
• Assignments made in two steps
  – each view-based learner proposes candidates, probabilities
  – Integrator combines evidence from multiple methods and constraints, assuming independent errors

M’ step:
• Each view-based learner re trains itself from the updated KB
• “context” methods create growing sets of contexts
NELL Architecture

Knowledge Base (latent variables)

Beliefs

Candidate Beliefs

Evidence Integrator

Text Context patterns (CPL)

HTML-URL context patterns (SEAL)

Morphology classifier (CML)

Learning and Function Execution Modules
Algorithm 1: Coupled Pattern Learner (CPL) Algorithm

Input: An ontology $\mathcal{O}$, and text corpus $C$

Output: Trusted instances/contextual patterns for each predicate

foreach predicate $p \in \mathcal{O}$ do

extract new candidate instances/contextual patterns using recently promoted patterns/instances;

filter candidates that violate coupling;

rank candidate instances/patterns;

promote top candidates;

end
learned extraction patterns: Company

retailers_like__ such_clients_as__ an_operating_business_of__ being_acquired_by__
firms_such_as__ a_flight_attendant_for__ chains_such_as__ industry_leaders_such_as__
advertisers_like__ social_networking_sites_such_as__ a_senior_manager_at__
competitors_like__ stores_like__ __is_an_ebay_company__ discounters_like__
a_distribution_deal_with__ popular_sites_like__ a_company_such_as__ vendors_such_as__
rivals_such_as__ competitors_such_as__ has_been_quoted_in_the__ providers_such_as__
company_research_for__ providers_like__ giants_such_as__ a_social_network_like__
popular_websites_like__ multinationals_like__ social_networks_such_as__
the_former_ceo_of__ a_software_engineer_at__ a_store_like__ video_sites_like__
a_social_networking_site_like__ giants_like__ a_company_like__ premieres_on__
corporations_such_as__ corporations_like__ professional_profile_on__ outlets_like__
the_executives_at__ stores_such_as__ __is_the_only_carrier__ a_big_company_like__
social_media_sites_such_as__ __has_an_article_today__ manufacturers_such_as__
companies_like__ social_media_sites_like__ companies__including__ firms_like__
networking_websites_such_as__ networks_like__ carriers_like__
social_networking_websites_like__ an_executive_at__ insured_via__ __provides_dialup_access
a_patent_infringement_lawsuit_against__ social_networking_sites_like__
social_network_sites_like__ carriers_such_as__ are_shipped_via__ social_sites_like__
a_licensing_deal_with__ portals_like__ vendors_like__ the_accounting_firm_of__
industry_leaders_like__ retailers_such_as__ chains_like__ prior_fiscal_years_for__
such_firms_as__ provided_free_by__ manufacturers_like__ airlines_like__ airlines_such_as__
learned extraction patterns: playsSport(arg1, arg2)

arg1_was_playing_arg2  arg2_megastar_arg1  arg2_icons_arg1  arg2_player_named_arg1
arg2_prodigy_arg1  arg1_is_the_tiger_woods_of_arg2  arg2_career_of_arg1
arg2_greats_as_arg1  arg1_plays_arg2  arg2_player_is_arg1  arg2_legends_arg1
arg1_announced_his_retirement_from_arg2  arg2_operations_chief_arg1  arg2_player_like_arg1
arg2_and_golfing_personalities_including_arg1  arg2_players_like_arg1  arg2_greats_like_arg1
arg2_players_are_steffi_graf_and_arg1  arg2_great_arg1  arg2_champ_arg1
arg2_greats_such_as_arg1  arg2_professionals_such_as_arg1  arg2_course_designed_by_arg1
arg2_hit_by_arg1  arg2_course_architects_including_arg1  arg2_greats_arg1  arg2_icon_arg1
arg2_stars_like_arg1  arg2_pros_like_arg1  arg1_retires_from_arg2  arg2_phenom_arg1
arg2_lesson_from_arg1  arg2_architects_robert_trent_jones_and_arg1  arg2_sensation_arg1
arg2_architects_like_arg1  arg2_pros_arg1  arg2_stars_venus_and_arg1
arg2_legends_arlne_palmer_and_arg1  arg2_hall_of_famer_arg1  arg2_racket_in_arg1
arg2_superstar_arg1  arg2_legend_arg1  arg2_legends_such_as_arg1  arg2_players_is_arg1
arg2_pro_arg1  arg2_player_was_arg1  arg2_god_arg1  arg2_idol_arg1
arg1_was_born_to_play_arg2  arg2_star_arg1  arg2_hero_arg1  arg2_course_architect_arg1
arg2_players_are_arg1  arg1_retired_from_professional_arg2  arg2_legends_as_arg1
arg2_autographed_by_arg1  arg2_related_quotations_spoken_by_arg1
arg2_courses_were_designed_by_arg1  arg2_player_since_arg1  arg2_match_between_arg1
arg2_course_was_designed_by_arg1  arg1_has_retired_from_arg2  arg2_player_arg1
arg1_can_hit_a_arg2  arg2_legends_including_arg1  arg2_player_than_arg1
arg2_legends_like_arg1  arg2_courses_designed_by_legends_arg1
arg2_player_of_all_time_is_arg1  arg2_fan_knows_arg1  arg1_learned_to_play_arg2
arg1_is_the_best_player_in_arg2  arg2_signed_by_arg1  arg2_champion_arg1
SEAL
Set Expander for Any Language
[Wang and Cohen, 2007]

Seeds

ford, toyota, nissan

Extraction

<li class="ford"><a href="http://www.curryauto.com/"/>

... 

<li class="honda"><a href="http://www.curryauto.com/"/>

... 

<li class="nissan"><a href="http://www.curryauto.com/"/>

... 

<li class="toyota"><a href="http://www.curryauto.com/"/>

...
Incorporating SEAL

For each category and relation being learned,

- Call Google search for sample of positive instances
- Learn URL-specific extractors for URL’s with multiple search hits

Typical learned SEAL extractors:

<table>
<thead>
<tr>
<th>Predicate</th>
<th>Web URL</th>
<th>Extraction Template</th>
</tr>
</thead>
<tbody>
<tr>
<td>academicField</td>
<td><a href="http://scholendow.ais.msu.edu/student/ScholSearch.Asp">http://scholendow.ais.msu.edu/student/ScholSearch.Asp</a></td>
<td>  [X] -</td>
</tr>
<tr>
<td>bird</td>
<td><a href="http://www.michaelforsberg.com/stock.html">http://www.michaelforsberg.com/stock.html</a></td>
<td></td>
</tr>
<tr>
<td>bookAuthor</td>
<td><a href="http://lifebehindthecurve.com/">http://lifebehindthecurve.com/</a></td>
<td></td>
</tr>
</tbody>
</table>
CMC: Morphology Learner

[Burr Settles]

- Logistic regression classifier per predicate
- Only trained for predicates with 100 positive examples
- Negative examples from constraint propagation

<table>
<thead>
<tr>
<th>Predicate</th>
<th>Feature</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>mountain</td>
<td>LAST=peak</td>
<td>1.791</td>
</tr>
<tr>
<td>mountain</td>
<td>LAST=mountain</td>
<td>1.093</td>
</tr>
<tr>
<td>mountain</td>
<td>FIRST=mountain</td>
<td>-0.875</td>
</tr>
<tr>
<td>musicArtist</td>
<td>LAST=band</td>
<td>1.853</td>
</tr>
<tr>
<td>musicArtist</td>
<td>POS=DT_NNS</td>
<td>1.412</td>
</tr>
<tr>
<td>musicArtist</td>
<td>POS=DT_JJ_NN</td>
<td>-0.807</td>
</tr>
<tr>
<td>newspaper</td>
<td>LAST=sun</td>
<td>1.330</td>
</tr>
<tr>
<td>newspaper</td>
<td>LAST=university</td>
<td>-0.318</td>
</tr>
<tr>
<td>newspaper</td>
<td>POS=NN_NNS</td>
<td>-0.798</td>
</tr>
<tr>
<td>university</td>
<td>LAST=college</td>
<td>2.076</td>
</tr>
<tr>
<td>university</td>
<td>PREFIX=uc</td>
<td>1.999</td>
</tr>
<tr>
<td>university</td>
<td>LAST=state</td>
<td>1.992</td>
</tr>
<tr>
<td>university</td>
<td>LAST=university</td>
<td>1.745</td>
</tr>
<tr>
<td>university</td>
<td>FIRST=college</td>
<td>-1.381</td>
</tr>
<tr>
<td>visualArtMovement</td>
<td>SUFFIX=ism</td>
<td>1.282</td>
</tr>
<tr>
<td>visualArtMovement</td>
<td>PREFIX=journ</td>
<td>-0.234</td>
</tr>
<tr>
<td>visualArtMovement</td>
<td>PREFIX=budd</td>
<td>-0.253</td>
</tr>
</tbody>
</table>
**Coupled Training Helps!**

[Carlson et al., WSDM 2010]

Using only two views: Text, HTML contexts.

<table>
<thead>
<tr>
<th>PRECISION</th>
<th>Text</th>
<th>HTML</th>
<th>Coupled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Categories</td>
<td>.41</td>
<td>.59</td>
<td>.90</td>
</tr>
<tr>
<td>Relations</td>
<td>.69</td>
<td>.91</td>
<td>.95</td>
</tr>
</tbody>
</table>

10 iterations, 200 M web pages. 44 categories, 27 relations. 199 extractions per category.
If coupled learning is the key idea, how can we get new coupling constraints?
Key Idea 2:
Discover New Coupling Constraints

- first order, probabilistic horn clause constraints

\[
0.93 \text{ athletePlaysSport(} ?x, ?y \text{) } \leftarrow \text{ athletePlaysForTeam(} ?x, ?z \text{) }
\]
\[
\text{ teamPlaysSport(} ?z, ?y \text{) }
\]

- connects previously uncoupled relation predicates

- infers new beliefs for KB
Discover New Coupling Constraints

For each relation:

- seek probabilistic first order Horn Clauses

- Positive examples: extracted beliefs in the KB
- Negative examples: ???
  - constraints don’t provide type-consistent negatives

Ontology to the rescue:

- numberOfValues(teamPlaysSport) = 1
- numberOfValues(competesWith) = any

(can infer negative examples from positive for this, but not for this)
Example Learned Horn Clauses

0.95  \( \text{athletePlaysSport}(\texttt{x}, \texttt{basketball}) \leftarrow \text{athleteInLeague}(\texttt{x}, \texttt{NBA}) \)

0.93  \( \text{athletePlaysSport}(\texttt{x}, \texttt{y}) \leftarrow \text{athletePlaysForTeam}(\texttt{x}, \texttt{z}) \\
      \quad \text{teamPlaysSport}(\texttt{z}, \texttt{y}) \)

0.91  \( \text{teamPlaysInLeague}(\texttt{x}, \texttt{NHL}) \leftarrow \text{teamWonTrophy}(\texttt{x}, \texttt{Stanley\_Cup}) \)

0.90  \( \text{athleteInLeague}(\texttt{x}, \texttt{y}) \leftarrow \text{athletePlaysForTeam}(\texttt{x}, \texttt{z}), \\
      \quad \text{teamPlaysInLeague}(\texttt{z}, \texttt{y}) \)

0.88  \( \text{cityInState}(\texttt{x}, \texttt{y}) \leftarrow \text{cityCapitalOfState}(\texttt{x}, \texttt{y}), \text{cityInCountry}(\texttt{y}, \texttt{USA}) \)

0.62* \( \text{newspaperInCity}(\texttt{x}, \texttt{New\_York}) \leftarrow \text{companyEconomicSector}(\texttt{x}, \texttt{media}) \\
      \quad \text{generalizations}(\texttt{x}, \texttt{blog}) \)
Some rejected learned rules

\[ \text{teamPlaysInLeague}(?x \ nba) \leftarrow \text{teamPlaysSport}(?x \ \text{basketball}) \ \text{0.94} \ [35 \ 0 \ 35] \ \text{[positive negative unlabeled]} \]

\[ \text{cityCapitalOfState}(?x \ ?y) \leftarrow \text{cityLocatedInState}(?x \ ?y), \text{teamPlaysInLeague}(?y \ nba) \ \text{0.80} \ [16 \ 2 \ 23] \]

\[ \text{teamplayssport}(?x, \ \text{basketball}) \leftarrow \text{generalizations}(?x, \ \text{university}) \ \text{0.61} \ [246 \ 124 \ 3063] \]
Rule Learning Summary

• Rule learner run every 10 iterations
• Manual filtering of rules

• After 120 iterations
  – 565 learned rules
  – 86% survived manual filter
  – 3948 new beliefs inferred by 486 surviving rules

• Effectiveness limited by sparsity of relations in ontology, and restriction on numberOfValues(R)=1
Learned Probabilistic Horn Clause Rules

0.93 \text{playsSport}(\text{?x,?y}) \leftarrow \text{playsForTeam}(\text{?x,?z}), \text{teamPlaysSport}(\text{?z,?y})
NELL Architecture, June 2010

- Learning and Function Execution Modules
  - Knowledge Base (latent variables)
    - Beliefs
    - Candidate Beliefs
    - Evidence Integrator
  - Text Context patterns (CPL)
  - HTML-URL context patterns (SEAL)
  - Morphology classifier (CML)
  - Rule Learner (RL)
NELL – June 2010

• 304,000 assertions
• ~30,000 learned text extraction patterns
• 486 accepted learned rules → 3948 new assertions
• Human check/clean KB every 10 iterations, beginning with iteration 100
• 65-75% of predicates currently populating well, others receiving significant correction
NELL Lessons

• Coupled semi-supervised learning of many functions helps!

• Learn new coupling constraints over time
NELL Lessons

- Coupled semi-supervised learning of many functions helps!
- Learn new coupling constraints over time
- We’ve changed the accuracy vs. experience learning curve from ___ to ___
  but not to ___
NELL – Next Steps
Distinguish Text Tokens from Entities

Coreference Resolution:
- Co-train classifier to predict coreference as $f(\text{string similarity, extracted beliefs})$
- Small amount of supervision: ~10 labeled coreference decisions
- Cluster tokens using $f$ as similarity measure

[Jayant Krishnamurthy]
Preliminary Coreference Results

- Evaluation on “sportsteam” category
- 90% precision, 61% recall for coreference decisions

<table>
<thead>
<tr>
<th>“sportsteam” Entities</th>
<th>Referring Tokens</th>
</tr>
</thead>
<tbody>
<tr>
<td>St. Louis Rams</td>
<td>st_louis Rams, louis Rams, st___louis Rams, Rams, st___louis Rams</td>
</tr>
<tr>
<td>Stanford Cardinals</td>
<td>stanford University, stanford Cardinals, Stanford</td>
</tr>
<tr>
<td>Pittsburgh Pirates</td>
<td>pittsburgh_pirates, Pirates, pittsburg_pirates</td>
</tr>
<tr>
<td>Los Angeles Lakers</td>
<td>lakers, la_lakers, los_angeles_lakers</td>
</tr>
<tr>
<td>Valdosta State Blazers</td>
<td>valdosta_blazers, valdosta_st__blazers, valdosta_state_blazers</td>
</tr>
<tr>
<td>Illinois State</td>
<td>illinois_state, illinois_state_university, illinois_university</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Ontology Extension

Idea:

• Discover frequently stated relations among ontology categories

• Given categories C1, C2, cluster pairs of known instances by their text contexts

* additional experiments with Etzioni & Soderland using TextRunner
## Preliminary Results

[Mohamed & Hruschka]

<table>
<thead>
<tr>
<th>Category Pair</th>
<th>Name</th>
<th>Text contexts</th>
<th>Proposed Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>MusicInstrument</td>
<td>Master</td>
<td>ARG1 master ARG2 ARG1 virtuoso ARG2 ARG1 legend ARG2 ARG2 plays ARG1</td>
<td>sitar, George Harrison tenor sax, Stan Getz trombone, Tommy Dorsey vibes, Lionel Hampton</td>
</tr>
<tr>
<td>Musician</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Disease</td>
<td>IsDueTo</td>
<td>ARG1 is due to ARG2 ARG1 is caused by ARG2</td>
<td>pinched nerve, herniated disk tennis elbow, tendonitis blepharospasm, dystonia</td>
</tr>
<tr>
<td>Disease</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CellType Chemical</td>
<td>ThatRelease</td>
<td>ARG1 that release ARG2 ARG2 releasing ARG1</td>
<td>epithelial cells, surfactant neurons, serotonin mast cells, histamine</td>
</tr>
<tr>
<td>Mammals Plant</td>
<td>Eat</td>
<td>ARG1 eat ARG2 ARG2 eating ARG1</td>
<td>koala bears, eucalyptus sheep, grasses goats, saplings</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Active Learning through CrowdSourcing
[Edith Law, Burr Settles, Luis von Ahn]

mockup of Polarity

Two-person game to collect:
• Labels for NP’s
• Information on multiple word senses and ambiguities
What will move forward research on Never Ending Learning?
Never Ending Learning: Thesis topics 1

Case study theses:

- office robot
- softbots
  - Web based research assistant
- game players
  - Why isn’t there a never-ending chess learner?
- never-ending learners for sensors
  - intelligent street corner camera
  - intelligent traffic control light
  - intelligent traffic grid
Never Ending Learning: Thesis topics 2

• Scaling EM: billions of virtual(?) latent variables
  – convergence properties?
  – what properties of constraint graph predict success?

• How are correctness and self-consistency related?
  – disagreement bounds error when functions co-trained on
    conditionally independent features [Dasgupta, et al., 2003]

• Curriculum-based learning
  – what curriculum properties guarantee improved long term
    learning?

• Self-reflection:
  – what self-reflection and self-repairing capabilities assure
    “reachability” of target performance?
thank you!

and thanks to Yahoo! for M45 computing
and thanks to Google, NSF, DARPA for partial funding