Information extraction: Reading the Web

Vivi Nastase

Summer semester 2012, ICL, University of Heidelberg

Never Ending Language Learning (NELL)

Research Goal

A never-ending machine learning system for extracting structured information from unstructured Web pages. The end result should be a knowledge base that reflects the content of the Web.

http://rtw.ml.cmu.edu/rtw/overview

NELL approach

Input

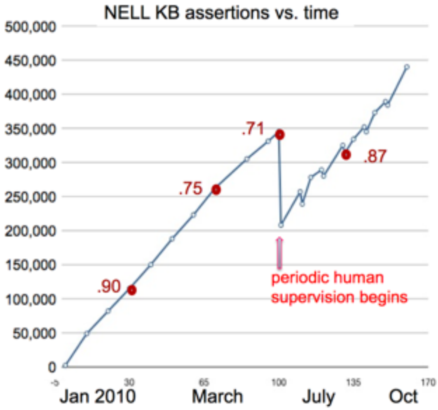
1. ontology with hundreds of categories (e.g. *person*, *sportsTeam*, *emotion*) and relations (e.g. *playsOnTeam(athlete,sportsTeam)*, *playsInstrument(musician,instrument)*) that NELL is expected to read about.
2. 10-15 examples of each category and relation
3. data
   - collection of 500 million pages
   - access to the rest of the Web

Process

- extract new instances of categories and relations to further populate a growing knowledge base of structured facts and knowledge
- learn to read better than the day before – from previous day's text sources, extract more information, more accurately

Assumptions

- Rely on redundancy of information on the Web – using different learning methods to extract complementary facets of this data
- Retrain using human feedback on the most blatant errors



Learning

Learning two types of knowledge

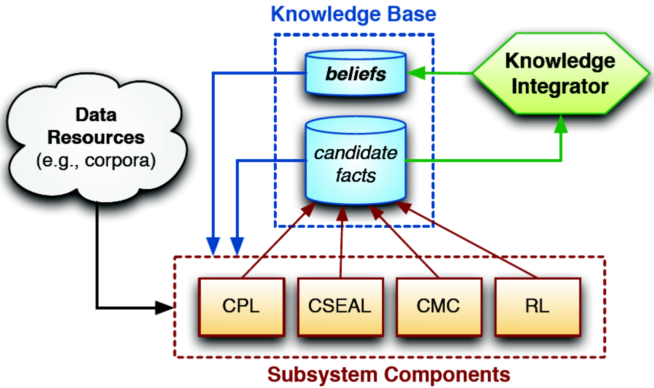
Learn categories which noun phrases refer to which semantic categories (e.g. *cities, companies, teams*)

Learn relations which pairs of noun phrases satisfy which semantic relations (e.g. *hasOfficesIn(organization, location)*)

Approach

- use free-form text patterns for extracting knowledge from sentences
- learn to extract knowledge from semi-structured web data (e.g. tables, lists)
- learn morphological regularities of instances of categories
- learn probabilistic Horn clause rules for inferring new instances of relations from already learned relations

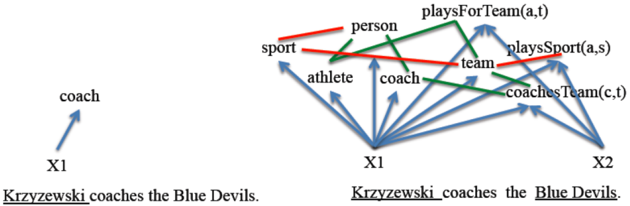
NELL architecture overview



Coupled semi-supervised learning for IE

Carlson et al., 2010 *Coupled semi-supervised learning for information extraction*

Semi-supervised learning – a small number of labeled examples, a large volume of unannotated text.



(A) A difficult semi-supervised learning problem

(B) An easier semi-supervised learning problem

Significant improvements come from coupling the training of information extractors for many interrelated categories and relations (B), compared with the task of learning a single information extractor (A).

Issues with bootstrapping

Semantic drift:

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

Coupled training

1. train classifiers using a small amount of labeled data
2. use the classifiers to label unlabeled data
3. the most confident new labels are added to the pool of data used to train the models

Coupling constraints for restricting allowable candidates

- output constraints: *mutual exclusion* – mutually exclusive predicates cannot both be satisfied by the same input x
- compositional constraints: *relation argument type checking* – the arguments of a relation to be learned must be of pre-declared types
- multi-view agreement constraints: *unstructured and semi-structured text features* – freeform textual context / HTML tags

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

Subsystem components

These components each assign a probability for each proposed candidate, and a summary of its evidence.

- use subsystem components that make uncorrelated errors
- learn multiple types of inter-related knowledge:
  - learn predicates from texts
  - learn to infer new relations from learned relations
- use coupled semi-supervised learning methods to leverage constraints between predicates being learned
  - categories and relations taxonomy (set-subset relations, mutually exclusive categories, categories as relations' expected arguments)
- distinguish high-confidence beliefs in the KB from lower-confidence candidates
- use a uniform KB representation to capture candidate facts and promoted beliefs of all types

Subsystem components – CPL

Algorithm 1: Coupled Pattern Learner (CPL)

Input: An ontology $\mathcal{O}$, and text corpus C
Output: Trusted instances/contextual patterns for each predicate

for $i = 1, 2, \ldots, \infty$ **do**
  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
end

Subsystem components – CSEAL

Algorithm 2: Coupled SEAL (CSEAL)

Input: An ontology $\mathcal{O}$, and text corpus C
Output: Trusted instances/wrappers for each predicate

```
for i = 1, 2, ..., ∞ do
    foreach predicate p ∈ O do
        begin Call existing SEAL code to:
            QUERY for documents containing recently
            promoted instances;
            LEARN wrappers for each document returned;
            EXTRACT new candidates using wrappers;
        end
        FILTER wrappers that extract candidates that
        violate coupling;
        RANK candidate instances;
        PROMOTE top candidates;
    end
end
```

Subsystem components – CMC and RL

Coupled Morphological Classifier

- classify NPs based on morphological features (words, capitalizations, affixes, POS, etc.)
- it applies to predicates that have at least 100 (promoted) instances
- uses mutually exclusion relationships to identify negative instances

Rule Learner

- first-order relational learner – learns probabilistic Horn clauses
  $\mathit{athletePlaysSport}(x, y) \leftarrow$
  $\mathit{athletePlaysForTeam}(x, z) \wedge \mathit{teamPlaysSport}(z, y)$
- these rules are used to infer new relation instances from relation instances already in the KB
- connects previously uncoupled relation predicates

Extracted predicates

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

More here: `http://rtw.ml.cmu.edu/rtw/`

Ontology extension

Goal

- Discover frequently stated relations among ontology categories
- Discover category subcategories

Approach

- For each pair of categories: co-cluster pairs of known instances and the contexts that connect them.
- when subclasses are extracted instead of instances, add subclass

Discovered relations

Category Pair	Name	Text contexts	Extracted Instances
MusicInstrument Musician	Master	ARG1 master ARG2 ARG1 virtuoso ARG2 ARG1 legend ARG2 ARG2 plays ARG1	sitar , George Harrison tenor sax, Stan Getz trombone, Tommy Dorsey vibes, Lionel Hampton
Disease Disease	IsDueTo	ARG1 is due to ARG2 ARG1 is caused by ARG2	pinched nerve, herniated disk tennis elbow, tendonitis blepharospasm, dystonia
CellType Chemical	ThatRelease	ARG1 that release ARG2 ARG2 releasing ARG1	epithelial cells, surfactant neurons, serotonin mast cells, histomine
Mammals Plant	Eat	ARG1 eat ARG2 ARG2 eating ARG1	koala bears, eucalyptus sheep, grasses goats, saplings
...			

Discovered subcategories

Original Category	SubType discovered by reading	Extracted Instances
Chemical	Gases	amonia, carbon_dioxide, carbon_monoxide, methane, sulphur, oxides, nitrous_oxides, water_vapor, ozone, nitrogen
Animal	LiveStock	chickens, cows, sheep, goats, pigs
Profession	Professionals	surgeons, chiropractors, dentists, engineers, medical staff, midwives, professors, scientists, specialists, technologists, aides

NELL now

Approx. 15 million candidate beliefs, 988,332 with high confidence.

Recently-Learned Facts twitter

Refresh

instance	iteration	date learned	confidence
association_of_america_s_public_tv_stations is a professional organization	568	14-may-2012	93.9
n1996_cricket_world_cup is a sporting event	572	20-may-2012	91.6
kelvin_sampson coaches a sports team	569	15-may-2012	99.8
kevin_wang is an author in the scientific field of machine learning	568	14-may-2012	92.4
the_benefactor is a TV show	569	15-may-2012	99.0
system is a subpart of the body within colon	572	20-may-2012	99.8
jaguar is a specific automobile maker dealer in houston	572	20-may-2012	100.0
basketball is a sport played in the venue american_airlines_center	571	18-may-2012	96.9
general_motors is a company in the economic sector of manufacturing	572	20-may-2012	96.9
milwaukee_bucks is a sports team that plays the sport basketball	572	20-may-2012	99.4

Open Information Extraction at Web Scale: Machine Reading for KnowItAll

Oren Etzioni, Turing Center, University of Washington

Reading the Web

Human Reading	**Machine Reading**
■ High precision	■ Noisy
■ Broad scope	■ Limited scope
■ Sentence-by-sentence	■ **Corpus-wide statistics**
■ High comprehension	■ Minimal reasoning
■ Background Knowledge.	■ Bottom up
■ Single language	■ **General**
■ Slow	■ **Very Fast!**

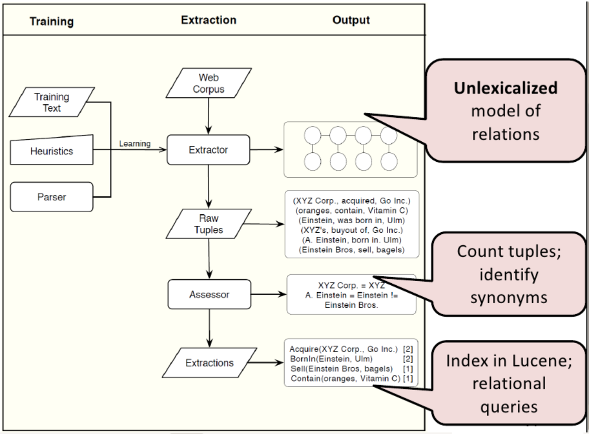
Open vs. traditional IE

	Traditional IE	Open IE
Input:	Corpus + **hand labeled data**	Corpus/Web + existing reso
Relations:	**Specified in advance**	Discovered automaticall
Complexity:	O(D x **R**)	O(D)
Output:	relation-specific	relation independent

Extraction on a large scale

Banko et al., 2007 *Open information extraction from the Web*

Distant supervision ➔ **180,000** training examples



TextRunner special features

1. self-supervised learner
2. single-pass extractor
3. redundancy-based assessor

Self-supervised learner

Input A small corpus sample

Process

1. automatically label training data as positive/negative:
   - find all base NPs: e_i
   - for each $(e_i, e_j), i < j$ – extract the grammatical relation path between them as potential relation r_{ij}
   - label $t = (e_i, r_{ij}, e_j)$ as positive if r_{ij} fulfill certain constraints (length, locality, type of e_i, e_j)
2. use labeled data to train a Naive Bayes classifier using domain independent features (later approaches – CRF):
   - the presence of POS tag sequences in r_{ij},
   - nr. of tokens in r_{ij},
   - nr. of stopwords in r_{ij},
   - whether e_i/e_j is a proper noun,
   - the POS to the left of e_i,
   - the POS to the right of e_j

Output relation tuples $t = (e_i, r_{ij}, e_j)$

Single-pass extractor

- one pass over the (large) corpus
- POS tag (most probable POS tag for each word)
- chunking for identifying NPs
- build candidate tuples (discard PPs, adverbs, etc)
  was originally developed by → *was developed by*
  Scientists from many university are studying ... → *Scientists are studying ...*
- represent candidate tuples through the features defined for the SSL, and feed them to the classifier

Redundancy-based assessor

assign a probability to each tuple t to express a certain relation based on the number of distinct sentences from which it was extracted (relations were normalized):
t appears k times in n sentences that match a clue:

$$P(t \in C | k, n) = \frac{\sum_{r \in num(C)} (\frac{r}{s})^k (1 - \frac{r}{s})^{n-k}}{\sum_{r' \in num(C \cup E)} (\frac{r'}{s})^k (1 - \frac{r'}{s})^{n-k}}$$

- C – set of unique target labels
- E – set of unique error labels ($num(E)$ also Zipf distributed
- $num(b)$ – the function giving the number of instances labeled $b \in C \cup E$
- $num(C)$ – the multi-set giving the number of intances for each label b $num(C)$ – Zipf distributed: if c_i is the i^{th} most frequently repeated label in C, $num(c_i) \propto i^{-z_C}$ (z_C is the parameter of the curve)
- s is the total number of instances

Error analysis

Incoherence relations (13%)

Sentence	Incoherent relation
The guide *contains* dead links and *omits* sites.	contains omits
The Mark 14 *was central* to the *torpedo* scandal of the fleet.	was central torpedo
They *recalled* that Nungesser *began* his career as a precinct leader	recalled began

Uninformative relations (7%)

Relation	Examples
is	… *is* an album by …, … *is* the author of …
has	… *has* a population of …, … *has* a cameo in …
made	… *made* a deal with …, … *made* a promise to …
took	… *took* place in …, … *took* control over …
gave	… *gave* a talk at …, … *gave* new meaning to …
got	… *got* tickets to see …, … *got* funding for …

ReVerb

Fader et al., 2011 *Identifying relations for open information extraction*

relation phrases = phrases that express relations

Incoherent relations

the extracted phrase has no meaningful interpretation

... *was central* to the *torpedo* scandal ...

Remedy: syntactic and positional constraints

Uninformative relations

the extracted phrase contains only light verbs

... *is* the author of ...

Remedy: force a longer phrase by including nouns

Overly specific relations

is offering only modest greenhouse gas reduction targets at

Remedy: argument variation constraints – minimal number of different arguments

Identifying relations from verbs

1. Find longest phrase matching a syntactic constraint
   $(V|VW*P)$
   $V = verb$
   $W = (noun|adj|adv|pron|det)$
   $P = (prep|particle|inf.marker)$
2. Constraint on arguments:
   $|args(Relation)| > k$

ReVerb relation phrases

	Binary Verbal Relation Phrases
85%	Satisfy Constraints
8%	Non-Contiguous Phrase Structure Coordination: X is produced and maintained by Y Multiple Args: X was founded in 1995 by Y Phrasal Verbs: X turned Y off
4%	Relation Phrase Not Between Arguments Intro. Phrases: Discovered by Y, X ... Relative Clauses: ...the Y that X discovered
3%	Do Not Match POS Pattern Interrupting Modifiers: X has a lot of faith in Y Infinitives: X to attack Y

Relation extraction with ReVerb

Features and their weights for assigning a confidence score to extracted relations (logistic regression)

Weight	Feature
1.16	(x, r, y) covers all words in s
0.50	The last preposition in r is *for*
0.49	The last preposition in r is *on*
0.46	The last preposition in r is *of*
0.43	$len(s) \leq 10$ words
0.43	There is a WH-word to the left of r
0.42	r matches VW*P from Figure 1
0.39	The last preposition in r is *to*
0.25	The last preposition in r is *in*
0.23	10 words $< len(s) \leq$ 20 words
0.21	s begins with x
0.16	y is a proper noun
0.01	x is a proper noun
-0.30	There is an NP to the left of x in s
-0.43	20 words $< len(s)$
-0.61	r matches V from Figure 1
-0.65	There is a preposition to the left of x in s
-0.81	There is an NP to the right of y in s
-0.93	Coord. conjunction to the left of r in s

Filtering extractions by *interestingness*

Lin et al., *Identifying interesting assertions from the Web*

Informative facts: *... the FDA banned ephedra ...*

Less useful statements: *... the FDA banned products ...*

Interestingness

Depends on the domain:

- social media
  feedback (click data, comments, ...)
- automated mathematical discovery
  plausibility + novelty + surprisingness + comprehensibility + complexity
- databases/data mining
  unexpectedness

Interestingness in IE

- specific (vs. general) assertions
  Albert Einstein taught at Princeton
  vs. *Albert Einstein taught at a university*
  $\rightarrow$ prefer assertions that contain named entities
- distinguishing assertions
  Einstein was offered the presidency of Israel
  vs. *Einstein was a man*

$$\rightarrow AFOFRatio(E) = \frac{AssertionFrequency(E)}{ObjectFrequency(object(E)) + 1}$$

  take assertions E for which $1 < AFOFRation(E) \leq 10$
- basic (definitional) assertions
  assertions similar to those chosen by Wikipedia editors to be included in Wikipedia infoboxes

KnowItAll now