Information Extraction

William Wang

School of Computer Science
Carnegie Mellon University

yww@cs.cmu.edu

CIPS Summer School
07/25/2015
History of Summer School

1st MSRA Summer Workshop of Information Extraction:

June, 2005
IE Course Logistics

Don’t be afraid of asking questions!

Homepage:
http://www.cs.cmu.edu/~yww/ss2015.html

Prerequisite:
• No previous experience on IE is required.
• Some basic knowledge in Machine Learning.
Acknowledgement

Some of the slides are also adapted from Andrew McCallum, Sunita Sarawagi, Luke Zettlemoyer, Rion Snow, Pedro Domingos, Ralf Grishman, Raphael Hoffmann, and many other people.
Instructor

William Wang (CMU)

Teaching experience:
CMU Machine Learning (100+ students)
CMU Machine Learning for Large Dataset (60+ students)

Affiliations:
• Yahoo! Labs NYC (2015)
• Microsoft Research Redmond (2012-2013)
• Columbia University (2009-2011)
• University of Southern California (2010)
Research Interests

• machine learning
[Machine Learning 2015] [IJCAI 2015] [ACL 2015a] [CIKM 2014] [StarAI 2014] [CIKM 2013]

• natural language processing
[NAACL 2015a] [EMNLP 2014] [ACL 2014] [EMNLP 2013a] [EMNLP 2013b] [ACL 2012] [SIGDIAL 2012] [IJCNLP 2011] [COLING 2010]

• spoken language processing
[ACL 2015b] [NAACL 2015b] [INTERSPEECH 2015] [SLT 2014] [ASRU 2013] [ICASSP 2013] [CSL 2013] [SLT 2012] [ASRU 2011] [INTERSPEECH 2011] [SIGDIAL 2011] [Book Chapter 2011]
What is Information Extraction (IE)?

And why do we care?
Tsung-Dao Lee (T. D. Lee, Chinese: 李政道; pinyin: Lǐ Zhèngdào) (born November 24, 1926) is a Chinese-born American physicist, well known for his work on parity violation, the Lee Model, particle physics, relativistic heavy ion (RHIC) physics, nontopological solitons and soliton stars. He holds the rank of University Professor Emeritus at Columbia University, where he has taught since 1953 and from which he retired in 2012.[1]

In 1957, Lee, at the age of 30, won the Nobel Prize in Physics with C. N. Yang[2] for their work on the violation of parity law in weak interaction, which Chien-Shiung Wu experimentally verified.
Information Extraction

Definition:

extracting structured knowledge from unstructured or semi-structured data (e.g. free text and tables).

In this short course: we will focus on IE from text data.
October 14, 2002, 4:00 a.m. PT

For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

Today, Microsoft claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. Gates himself says Microsoft will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said Bill Veghte, a Microsoft VP. "That's a super-important shift for us in terms of code access."

Richard Stallman, founder of the Free Software Foundation, countered saying...
A Broader View of IE

As a family of techniques:

Information Extraction = segmentation + classification + association + clustering

October 14, 2002, 4:00 a.m. PT

For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

Today, Microsoft claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. Gates himself says Microsoft will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said Bill Veghte, a Microsoft VP. "That’s a super-important shift for us in terms of code access."

Richard Stallman, founder of the Free Software Foundation, countered saying...
A Broader View of IE

As a family of techniques:

Information Extraction = segmentation + classification + association + clustering

October 14, 2002, 4:00 a.m. PT

For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

Today, Microsoft claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. Gates himself says Microsoft will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said Bill Veghte, a Microsoft VP. "That's a super-important shift for us in terms of code access."

Richard Stallman, founder of the Free Software Foundation, countered saying...
A Broader View of IE

As a family of techniques:

Information Extraction = segmentation + classification + association + clustering

October 14, 2002, 4:00 a.m. PT

For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

Today, Microsoft claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. Gates himself says Microsoft will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said Bill Veghte, a Microsoft VP. "That's a super-important shift for us in terms of code access."

Richard Stallman, founder of the Free Software Foundation, countered saying…
A Broader View of IE

As a family of techniques:

Information Extraction = segmentation + classification + association + clustering

October 14, 2002, 4:00 a.m. PT

For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

Today, Microsoft claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. Gates himself says Microsoft will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said Bill Veghte, a Microsoft VP. "That's a super-important shift for us in terms of code access."

Richard Stallman, founder of the Free Software Foundation, countered saying…
Complexity in IE

Closed set

- U.S. states (50 states)
  - He was born in Alabama…
  - The big Wyoming sky…

Regular set

- U.S. phone numbers
  - Phone: (413) 545-1323
  - The CALD main office can be reached at 412-268-1299

Complex patterns

- U.S. postal addresses
  - University of Arkansas
    P.O. Box 140
    Hope, AR 71802
  - Headquarters:
    1128 Main Street, 4th Floor
    Cincinnati, Ohio 45210

Ambiguous patterns

- Person names
  - …was among the six houses sold by Hope Feldman that year.
  - Pawel Opalinski, Software Engineer at WhizBang Labs.
Granularity of IE Tasks

<table>
<thead>
<tr>
<th>Single entity</th>
<th>Binary relationship</th>
<th>N-ary record</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Person</strong>: Jack Welch</td>
<td><strong>Relation</strong>: Person-Title</td>
<td><strong>Relation</strong>: Succession</td>
</tr>
<tr>
<td><strong>Person</strong>: Jeffrey Immelt</td>
<td><strong>Person</strong>: Jack Welch</td>
<td><strong>Company</strong>: General Electric</td>
</tr>
<tr>
<td><strong>Title</strong>: CEO</td>
<td><strong>Title</strong>: CEO</td>
<td><strong>Title</strong>: CEO</td>
</tr>
<tr>
<td><strong>Out</strong>: Jack Welch</td>
<td><strong>In</strong>: Jeffrey Immelt</td>
<td><strong>Out</strong>: Jack Welch</td>
</tr>
<tr>
<td><strong>Location</strong>: Connecticut</td>
<td><strong>In</strong>: Jeffrey Immelt</td>
<td><strong>In</strong>: Jeffrey Immelt</td>
</tr>
<tr>
<td><strong>Location</strong>: Connecticut</td>
<td></td>
<td><strong>Location</strong>: Connecticut</td>
</tr>
</tbody>
</table>

Jack Welch will retire as CEO of General Electric tomorrow. The top role at the Connecticut company will be filled by Jeffrey Immelt.
IE Applications
Question Answering

where does td lee work

About 80,600,000 results (0.39 seconds)

He holds the rank of University Professor Emeritus at Columbia University, where he has taught since 1953 and from which he retired in 2012. In 1957, Lee, at the age of 30, won the Nobel Prize in Physics with C. N.

Tsung-Dao Lee - Wikipedia, the free encyclopedia
In 1957, Lee, at the age of 30, won the Nobel Prize in Physics with C. N. Yang for their work on the violation of parity law in weak interaction, which Chien-Shiung Wu experimentally verified. Lee was the youngest Nobel laureate after World War II until Malala Yousafzai was awarded the Nobel Peace Prize in 2014.

Tsung-Dao Lee - Wikipedia, the free encyclopedia
https://en.wikipedia.org/wiki/Tsung-Dao_Lee
Virtual Assistant

Apple Siri

Google Now

Windows Cortana
Course Outline

1. Basic theories and practices on named entity recognition: supervised, semi-supervised, unsupervised.

2. Recent advances in relation extraction:
   a. distant supervision
   b. latent variable models

3. Scalable IE and reasoning with first-order logics.
Basic Theories and Practices of NER
Named Entity Recognition

Given a sentence:

Yesterday William Wang flew to Beijing.

extract the following information:

Person name: William Wang
Location name: Beijing

What is the easiest method?

use a lexicon of person names and location names, scan the sentence and look for matches.

Why this will not work? The scalability issue.
Overview of NER Models

**Lexicons**
- Abraham Lincoln was born in Kentucky.
- Member?
- Alabama
- Alaska
- ...Wisconsin
- Wyoming

**Classify Pre-segmented Candidates**
- Abraham Lincoln was born in Kentucky.
- Classifier
- which class?

**Sliding Window**
- Abraham Lincoln was born in Kentucky.
- Classifier
- which class?
- Try alternate window sizes:

**Boundary Models**
- Abraham Lincoln was born in Kentucky.
- BEGIN
- END
- BEGIN
- END
- Classifier
- which class?

**Token Tagging**
- Abraham Lincoln was born in Kentucky.
- BEGIN
- END
- BEGIN
- END
- Most likely state sequence?

This is often treated as a structured prediction problem...classifying tokens *sequentially*

- HMMs, CRFs, ....
IE by Sliding Window

GRAND CHALLENGES FOR MACHINE LEARNING

Jaime Carbonell
School of Computer Science
Carnegie Mellon University

3:30 pm
7500 Wean Hall

Machine learning has evolved from obscurity in the 1970s into a vibrant and popular discipline in artificial intelligence during the 1980s and 1990s. As a result of its success and growth, machine learning is evolving into a collection of related disciplines: inductive concept acquisition, analytic learning in problem solving (e.g. analogy, explanation-based learning), learning theory (e.g. PAC learning), genetic algorithms, connectionist learning, hybrid systems, and so on.
GRAND CHALLENGES FOR MACHINE LEARNING

Jaime Carbonell
School of Computer Science
Carnegie Mellon University

3:30 pm
7500 Wean Hall

Machine learning has evolved from obscurity in the 1970s into a vibrant and popular discipline in artificial intelligence during the 1980s and 1990s. As a result of its success and growth, machine learning is evolving into a collection of related disciplines: inductive concept acquisition, analytic learning in problem solving (e.g. analogy, explanation-based learning), learning theory (e.g. PAC learning), genetic algorithms, connectionist learning, hybrid systems, and so on.
GRAND CHALLENGES FOR MACHINE LEARNING

Jaime Carbonell
School of Computer Science
Carnegie Mellon University

3:30 pm
7500 Wean Hall

Machine learning has evolved from obscurity in the 1970s into a vibrant and popular discipline in artificial intelligence during the 1980s and 1990s. As a result of its success and growth, machine learning is evolving into a collection of related disciplines: inductive concept acquisition, analytic learning in problem solving (e.g. analogy, explanation-based learning), learning theory (e.g. PAC learning), genetic algorithms, connectionist learning, hybrid systems, and so on.
Machine learning has evolved from obscurity in the 1970s into a vibrant and popular discipline in artificial intelligence during the 1980s and 1990s. As a result of its success and growth, machine learning is evolving into a collection of related disciplines: inductive concept acquisition, analytic learning in problem solving (e.g. analogy, explanation-based learning), learning theory (e.g. PAC learning), genetic algorithms, connectionist learning, hybrid systems, and so on.
A Naïve Bayes Sliding Window Model

[Freitag 1997]

Estimate Pr(LOCATION|window) using Bayes rule

Try all “reasonable” windows (vary length, position)

Assume independence for length, prefix words, suffix words, content words

Estimate from data quantities like: Pr(“Place” in prefix|LOCATION)

If P(“Wean Hall Rm 5409” = LOCATION) is above some threshold, extract it.
A Naïve Bayes Sliding Window Model

[Freitag 1997]

1. Create dataset of examples like these:
   +(prefix00,…,prefixColon, contentWean,contentHall,…..,suffixSpeaker,…)
   - (prefixColon,…,prefixWean,contentHall,…..,ContentSpeaker,suffixColon,…..)
   …

2. Train a NaiveBayes classifier (or YFCL), treating the examples like BOWs for text classification

3. If Pr(class=+|prefix,contents,suffix) > threshold, predict the content window is a location.
   • To think about: what if the extracted entities aren’t consistent, eg if the location overlaps with the speaker?

... 00 : pm Place : Wean Hall Rm 5409 Speaker : Sebastian Thrun ...

prefix | contents | suffix
Machine learning has evolved from obscurity in the 1970s into a vibrant and popular discipline in artificial intelligence during the 1980s and 1990s. As a result of its success and growth, machine learning is evolving into a collection of related disciplines: inductive concept acquisition, analytic learning in problem solving (e.g. analogy, explanation-based learning), learning theory (e.g. PAC learning), genetic algorithms, connectionist learning, hybrid systems, and so on.
Token Tagging
NER by Token Tagging

Given a sentence:

Yesterday William Wang flew to Beijing.

1) Break the sentence into tokens, and classify each token with a label indicating what sort of entity it’s part of:

Yesterday William Wang Wang flew to Beijing

2) Identify names based on the entity labels

Person name: William Wang
Location name: Beijing

3) To learn an NER system, use YFCL.
NER by Token Tagging

Similar labels tend to *cluster together* in text

Yesterday William Wang flew to Beijing

Another common labeling scheme is BIO (begin, inside, outside; e.g. beginPerson, insidePerson, beginLocation, insideLocation, outside)

BIO also leads to *strong dependencies between nearby labels* (eg inside follows begin)
Hidden Markov Models for NER

Given a sequence of observations:

Today William Wang is teaching at Peking University.

and a trained HMM:

Find the most likely state sequence: (Viterbi) \( \arg \max_{\bar{s}} P(\bar{s}, \bar{o}) \)

Any words said to be generated by the designated “person name” state extract as a person name:

Person name: William Wang
Review of Hidden Markov Models

\[ p(X, Z|\Theta) = p(z_1|\pi) \left[ \prod_{n=2}^{N} p(z_n|z_{n-1}, A) \right] \prod_{n=1}^{N} p(x_n|z_n, \phi) \]

Observables: \( X = \{x_1, \ldots, x_N\} \)
Latent states: \( Z = \{z_1, \ldots, z_N\} \)
Model parameters: \( \Theta = \{\pi, A, \phi\} \)
1. The HMM consists of two probability tables
   • $Pr(currentState=s|previousState=t)$ for s=background, location, speaker,
   • $Pr(currentWord=w|currentState=s)$ for s=background, location, ...

2. Estimate these tables with a (smoothed) CPT
   • $\text{Prob}(\text{location}|\text{location}) = \#(\text{loc}->\text{loc})/\#(\text{loc}->*)$ transitions

3. Given a new sentence, find the most likely sequence of hidden
   states using Viterbi method:
   $\text{MaxProb}(\text{curr}=s|\text{position } k) =$
   $\text{Max}_{\text{state } t} \text{MaxProb}(\text{curr}=t|\text{position}=k-1) \times \text{Prob}(\text{word}=w_{k-1}|t) \times \text{Prob}(\text{curr}=s|\text{prev}=t)$
Machine learning has evolved from obscurity in the 1970s into a vibrant and popular discipline in artificial intelligence during the 1980s and 1990s. As a result of its success and growth, machine learning is evolving into a collection of related disciplines: inductive concept acquisition, analytic learning in problem solving (e.g. analogy, explanation-based learning), learning theory (e.g. PAC learning), genetic algorithms, connectionist learning, hybrid systems, and so on.
Improving the HMMs

• we need richer representation for the observations e.g., overlapping features.

• we would like to consider modeling the discriminative/conditional probability model of $P(Z|X)$, rather than the joint/generative probability model of $P(Z,X)$. 
Maximum Entropy Markov Model (MEMM)
Naïve Bayes vs HMM

HMM = sequential Naïve Bayes
Replace generative model in HMM with a MaxEnt/Logistic Regression model
Why MaxEnt Model?

- **Performance:**
  Good MaxEnt methods are competitive with linear SVMs and other state of the art classifiers in accuracy.

- **Embedding in a larger system:**
  MaxEnt optimizes \( \Pr(y|x) \), not error rate.

- **Example: Text Categorization**

  (Zhang and Oles 2001)

  - Features are a *word* in document and *class* (they do feature selection to use reliable indicators)
  - Tests on classic Reuters data set (and others)
    - Naive Bayes: 77.0% \( F_1 \)
    - Linear regression: 86.0%
    - Logistic regression: 86.4%
    - Support vector machine: 86.5%
  - Emphasizes the importance of *regularization* (smoothing) for successful use of discriminative methods (not used in most early NLP/IR work)
From Naïve Bayes to MaxEnt

$$\Pr(y \mid x) = \frac{1}{Z} \Pr(y) \prod_j \Pr(w_k \mid y) = \alpha_0 \prod_i \alpha_i^{f_i(x)}$$

where $w_k$ is word $j$ in $x$

$$f_{j,k}(doc) = [\text{word } k \text{ appears at position } j \text{ of } doc?1:0]$$

$$f_i(doc) = i - \text{th } j, k \text{ combination}$$

$$\alpha_i = \Pr(w_k \mid y)$$

$$\alpha_0 = \Pr(y) / Z$$
MEMMs

• Basic difference from ME tagging:
  1. ME tagging: previous state is feature of MaxEnt classifier
  2. MEMM: build a separate MaxEnt classifier for each state.
     Can build any HMM architecture you want; eg parallel nested HMM’s, etc.
• MEMM does allow possibility of “hidden” states and Baum-Welsh like training
• Viterbi is the most natural inference scheme
MEMM task: FAQ parsing

X-NNTP-Poster: NewsHound v1.33
archive-name: acorn/faq/part2
Frequency: monthly

question>2.6) What configuration of serial cable should I use
answer> Here follows a diagram of the necessary connections
answer> programs to work properly. They are as far as I know the
answer> agreed upon by commercial comms software developers for
answer> Pins 1, 4, and 8 must be connected together inside
answer> is to avoid the well known serial port chip bugs. The
MEMM features

- begins-with-number
- begins-with-ordinal
- begins-with-punctuation
- begins-with-question-word
- begins-with-subject
- blank
- contains-alphanum
- contains-bracketed-number
- contains-http
- contains-non-space
- contains-number
- contains-pipe
- contains-question-mark
- contains-question-word
- ends-with-question-mark
- first-alpha-is-capitalized
- indented
- indented-1-to-4
- indented-5-to-10
- more-than-one-third-space
- only-punctuation
- prev-is-blank
- prev-begins-with-ordinal
- shorter-than-30
MEMM Performance

Table 4. Co-occurrence agreement probability (COAP), segmentation precision (SegPrec) and segmentation recall (SegRecall) of four learners on the FAQ dataset. All these averages have 95% confidence intervals of 0.01 or less.

<table>
<thead>
<tr>
<th>Learner</th>
<th>COAP</th>
<th>SegPrec</th>
<th>SegRecall</th>
</tr>
</thead>
<tbody>
<tr>
<td>ME-Stateless</td>
<td>0.520</td>
<td>0.038</td>
<td>0.362</td>
</tr>
<tr>
<td>TokenHMM</td>
<td>0.865</td>
<td>0.276</td>
<td>0.140</td>
</tr>
<tr>
<td>FeatureHMM</td>
<td>0.941</td>
<td>0.413</td>
<td>0.529</td>
</tr>
<tr>
<td>MEMM</td>
<td>0.965</td>
<td>0.867</td>
<td>0.681</td>
</tr>
</tbody>
</table>
Conditional Random Fields
Label Bias Problem of MEMM

- Consider a simple MEMM for person and location names
  - all names are two tokens states:
    - other
    - b-person and e-person for person names
    - b-locn and e-locn for location names
Label Bias Problem of MEMM

corpus:
Harvey Ford
(person 9 times, location 1 time)
Harvey Park
(location 9 times, person 1 time)
Myrtle Ford
(person 9 times, location 1 time)
Myrtle Park
(location 9 times, person 1 time)

second token a good indicator of person vs. location
Label Bias Problem of MEMM

*Conditional probabilities:*

$p(b\text{-}\text{person} \mid \text{other}, w = \text{Harvey}) = 0.5$

$p(b\text{-}\text{locn} \mid \text{other}, w = \text{Harvey}) = 0.5$

$p(b\text{-}\text{person} \mid \text{other}, w = \text{Myrtle}) = 0.5$

$p(b\text{-}\text{locn} \mid \text{other}, w = \text{Myrtle}) = 0.5$

$p(e\text{-}\text{person} \mid b\text{-}\text{person}, w = \text{Ford}) = 1$

$p(e\text{-}\text{person} \mid b\text{-}\text{person}, w = \text{Park}) = 1$

$p(e\text{-}\text{locn} \mid b\text{-}\text{locn}, w = \text{Ford}) = 1$

$p(e\text{-}\text{locn} \mid b\text{-}\text{locn}, w = \text{Park}) = 1$
Label Bias Problem of MEMM

Role of second token in distinguishing person vs. location completely lost
Label Bias Problem of MEMM

• Problem:
  Probabilities of outgoing arcs normalized separately for each state.
CRFs’ advantages

• over HMM: the independence assumption is relaxed, allowing overlapping features.
• over MEMM: undirected graphical model, a single exponential model for the joint probability of the entire label sequence.
Linear Chain CRFs

\[ p(y|x) = \frac{1}{Z(x)} \exp \sum_t \left( \sum_{k=1}^{K} \lambda_k f_k(y_t, y_{t-1}) + \sum_{h=1}^{h} \mu_h f_h(x_t, y_t) \right) \]
Sha & Pereira results

<table>
<thead>
<tr>
<th>$q(y_{i-1}, y_i)$</th>
<th>$p(x, t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y_i = y$</td>
<td>$y_i = y$, $y_{i-1} = y'$</td>
</tr>
<tr>
<td>$c(y_i) = c$</td>
<td>true</td>
</tr>
</tbody>
</table>

$w_i = w$
$w_{i-1} = w$
$w_{i+1} = w$
$w_{i-2} = w$
$w_{i+2} = w$
$w_{i-1} = w'$, $w_i = w$
$w_{i+1} = w'$, $w_{i+2} = w$
$t_i = t$
$t_{i-1} = t$
$t_{i+1} = t$
$t_{i-2} = t$
$t_{i+2} = t$
$t_{i-1} = t'$, $t_i = t$
$t_{i-2} = t'$, $t_{i-1} = t$
$t_i = t'$, $t_{i+1} = t$
$t_{i+1} = t'$, $t_{i+2} = t$
$t_{i-2} = t''$, $t_{i-1} = t'$, $t_i = t$
$t_{i-1} = t''$, $t_i = t'$, $t_{i+1} = t$
$t_i = t''$, $t_{i+1} = t'$, $t_{i+2} = t$

Table 1: Shallow parsing features

<table>
<thead>
<tr>
<th>Model</th>
<th>F score</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM combination</td>
<td>94.39%</td>
</tr>
<tr>
<td>(Kudo and Matsumoto, 2001)</td>
<td></td>
</tr>
<tr>
<td>CRF</td>
<td>94.38%</td>
</tr>
<tr>
<td>Generalized winnow</td>
<td>93.89%</td>
</tr>
<tr>
<td>(Zhang et al., 2002)</td>
<td></td>
</tr>
<tr>
<td>Voted perceptron</td>
<td>94.09%</td>
</tr>
<tr>
<td>MEMM</td>
<td>93.70%</td>
</tr>
</tbody>
</table>

Table 2: NP chunking F scores

CRF beats MEMM (McNemar’s test); MEMM probably beats voted perceptron
Sha & Pereira results

<table>
<thead>
<tr>
<th>training method</th>
<th>time</th>
<th>F score</th>
<th>$\mathcal{L}'$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precond. CG</td>
<td>130</td>
<td>94.19%</td>
<td>-2968</td>
</tr>
<tr>
<td>Mixed CG</td>
<td>540</td>
<td>94.20%</td>
<td>-2990</td>
</tr>
<tr>
<td>Plain CG</td>
<td>648</td>
<td>94.04%</td>
<td>-2967</td>
</tr>
<tr>
<td>L-BFGS</td>
<td>84</td>
<td>94.19%</td>
<td>-2948</td>
</tr>
<tr>
<td>GIS</td>
<td>3700</td>
<td>93.55%</td>
<td>-5668</td>
</tr>
</tbody>
</table>

Table 3: Runtime for various training methods in minutes, 375k examples
Sequential Models for IE: Practical Advice
Implementing an HMM

• Follow Larry Rabiner's classic HMM tutorial:

A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition

LAWRENCE R. RABINER, FELLOW, IEEE

• Debugging an HMM:
Training (forward-backward): check your transition probability matrix.
Decoding (Viterbi): check the output state sequence.
Understanding CRFs

• actually Lafferty’s paper is pretty hard to understand. Instead, try to read Hanna Wallach’s CRF introduction.

Conditional Random Fields: An Introduction*

Hanna M. Wallach

February 24, 2004
CRF Tools

• CRF++: probably most widely used. Fast, multithreaded L-BFGS training. Support CoNLL format only.

• CRFsuite: flexible data input format. No parallelization.

• Wapiti (recommended): Support CoNLL and customized data format. Fast, multithreaded L-BFGS training.

• Stochastic Gradient CRFs: using SGD training instead of L-BFGS.

• Mallet: CRFs in Java.
CRF Demo: Wapiti
https://wapiti.limsi.fr

Training sentence:
Yesterday William Wang flew to Beijing.

Testing sentence:
Yesterday William Cohen flew to Buenos Aires.
Semi-supervised IE
Semi-supervised IE

• Basic idea:
  Find where a known fact occurs in text, by matching/alignment/…
  Use this as training data for a conventional IE learning system.

• Once you’ve learned an extractor from that data
  Run the extractor on some (maybe additional) text
  Take the (possibly noisy) new facts and start over

• This is called: “Self-training” or “bootstrapping”
Macro-reading c. 1992

Automatic Acquisition of Hyponyms from Large Text Corpora

Marti A. Hearst
Computer Science Division, 571 Evans Hall
University of California, Berkeley
Berkeley, CA 94720
and
Xerox Palo Alto Research Center
marti@cs.berkeley.edu

Idea: write some specific patterns that indicate A is a kind of B:

1. ... such NP as NP (“at such schools as CMU, students rarely need extensions”)

2. NP, NP, or other NP (“William, Carlos or other machine learning professors”)

3. NP including NP (“struggling teams including the Pirates”)

4. NP, especially NP (prestigious conferences, especially NIPS)

[Coling 1992]

Results: 8.6M words of Grolier’s encyclopedia → 7067 pattern instances → 152 relations

Many were not in WordNet.

Marti’s system was iterative
Another iterative, high-precision system

Extracting Patterns and Relations from the World Wide Web

Sergey Brin
Computer Science Department
Stanford University
sergey@cs.stanford.edu

[some workshop, 1998]

Unlike Hearst, Brin learned the patterns; and learned very high-precision, easy-to-match patterns using regular expressions.

Result: 24M web pages + 5 books $\rightarrow$ 199 occurrences $\rightarrow$ 3 patterns $\rightarrow$ 4047 occurrences + 5M pages $\rightarrow$ 3947 occurrences $\rightarrow$ 105 patterns $\rightarrow$ ... 15,257 books *with some manual tweaks

Idea: exploit “pattern/relation duality”:

1. Start with some seed instances of (author,title) pairs (“Isaac Asimov”, “The Robots of Dawn”)
2. Look for occurrences of these pairs on the web.
3. Generate patterns that match the seeds.
   - URLprefix, prefix, middle, suffix
4. Extract new (author, title) pairs that match the patterns.
5. Go to 2.
Key Ideas: So Far

- High-precision low-coverage extractors and large redundant corpora (macro-reading)
- Self-training/bootstrapping
  1) Advantage: train on a small corpus, test on a larger one
     - You can use more-or-less off-the-shelf learning methods
     - You can work with very large corpora
  2) But, data gets noisier and noisier as you iterate
  3) Need either
     - really high-precision extractors, or
     - some way to cope with the noise
A variant of bootstrapping: co-training

Redundantly Sufficient Features:

- features $x$ can be separated into two types $x_1, x_2$
- either $x_1$ or $x_2$ is sufficient for classification – i.e.
  there exists functions $f_1$ and $f_2$ such that
  \[ f(x) = f_1(x_1) = f_2(x_2) \text{ has low error} \]

.., says Mr. Cooper, a vice president of..

- spelling feature e.g. Capitalization=$X+.X+$
  Prefix=$Mr.$
- context feature e.g., based on words nearby in dependency parse
Another kind of self-training

Combining Labeled and Unlabeled Data with Co-Training

Avrim Blum
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213-3891
avrism@cs.cmu.edu

Tom Mitchell
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213-3891
mitchell@cs.cmu.edu

Given:

- a set $L$ of labeled training examples
- a set $U$ of unlabeled examples

Create a pool $U'$ of examples by choosing $u$ examples at random from $U$

Loop for $k$ iterations:

1. Use $L$ to train a classifier $h_1$ that considers only the $x_1$ portion of $x$
2. Use $L$ to train a classifier $h_2$ that considers only the $x_2$ portion of $x$
3. Allow $h_1$ to label $p$ positive and $n$ negative examples from $U'$
4. Allow $h_2$ to label $p$ positive and $n$ negative examples from $U'$
5. Add these self-labeled examples to $L$
6. Randomly choose $2p + 2n$ examples from $U$ to replenish $U'$

Figure 1: Graphs $G_D$ and $G_S$. Edges represent examples with non-zero probability under $D$. Solid edges represent examples observed in some finite sample $S$. Notice that given our assumptions, even without seeing any labels the learning algorithm can deduce that any two examples belonging to the same connected component in $G_S$ must have the same classification.
A co-training algorithm

Given:

- a set $L$ of labeled training examples
- a set $U$ of unlabeled examples

Create a pool $U'$ of examples by choosing $u$ examples at random from $U$

Loop for $k$ iterations:

- Use $L$ to train a classifier $h_1$ that considers only the $x_1$ portion of $x$
- Use $L$ to train a classifier $h_2$ that considers only the $x_2$ portion of $x$
- Allow $h_1$ to label $p$ positive and $n$ negative examples from $U'$
- Allow $h_2$ to label $p$ positive and $n$ negative examples from $U'$
- Add these self-labeled examples to $L$
- Randomly choose $2p + 2n$ examples from $U$ to replenish $U'$
Unsupervised Models for Named Entity Classification
Michael Collins and Yoram Singer [EMNLP 99]

Redundantly Sufficient Features:
- features $x$ can be separated into two types $x_1, x_2$
- either $x_1$ or $x_2$ is sufficient for classification – i.e.

There exists functions $f_1$ and $f_2$ such that

$$f(x) = f_1(x_1) = f_2(x_2)$$

has low error

Candidate entities $x$ segmented using a POS pattern

.., says Mr. Cooper, a vice president of ..

spelling feature

context feature

e.g., Capitalization=X+.X+
Prefix=Mr.

Based on dependency parse
Evaluation for Collins and Singer

88,962 examples (spelling, context) pairs
7 seed rules are used
1000 examples are chosen as test data (85 noise)
We label the examples as (location, person, organization, noise)

<table>
<thead>
<tr>
<th>Learning Algorithm</th>
<th>Accuracy (Clean)</th>
<th>Accuracy (Noise)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>45.8%</td>
<td>41.8%</td>
</tr>
<tr>
<td>EM</td>
<td>83.1%</td>
<td>75.8%</td>
</tr>
<tr>
<td>(Yarowsky 95)</td>
<td>81.3%</td>
<td>74.1%</td>
</tr>
<tr>
<td>Yarowsky-cautious</td>
<td>91.2%</td>
<td>83.2%</td>
</tr>
<tr>
<td>DL-CoTrain</td>
<td>91.3%</td>
<td>83.3%</td>
</tr>
<tr>
<td>CoBoost</td>
<td>91.1%</td>
<td>83.1%</td>
</tr>
</tbody>
</table>

Table 2: Accuracy for different learning methods. The baseline method tags all entities as the most frequent class type (organization).

\[
\text{Accuracy : Noise} = \frac{N_c}{962}
\]

\[
\text{Accuracy : Clean} = \frac{N_c}{962 - 85}
\]
Key Ideas: So Far

- High-precision low-coverage extractors and large redundant corpora (macro-reading)
- Self-training/bootstrapping
- Co-training
- Clustering phrases by context

Don’t propagate labels;
Instead do without them entirely
Basic idea: parse a big corpus, then cluster NPs by their contexts
Key Ideas: So Far

• High-precision low-coverage extractors and large redundant corpora (macro-reading)
• Self-training/bootstrapping or co-training
• Other semi-supervised methods:
  1) Expectation-maximization: like self-training but you “soft-label” the unlabeled examples with the expectation over the labels in each iteration.
  2) Works for almost any generative model (e.g., HMMs)
  3) Learns directly from all the data
     Maybe better; Maybe slower
     Extreme cases:
     supervised learning …. clustering + cluster-labeling
Key Ideas: So Far

• High-precision low-coverage extractors and large redundant corpora (macro-reading)
• Self-training/bootstrapping or co-training
• Other semi-supervised methods:
  - Expectation-maximization
  - Transductive margin-based methods (e.g., transductive SVM)
  - Graph-based methods
History: Open-domain IE by pattern-matching (Hearst, 92)

- Start with seeds: “NIPS”, “ICML”
- Look thru a corpus for certain patterns:
  - … “at NIPS, AISTATS, KDD and other learning conferences.”
- Expand from seeds to new instances

Repeat….until ___
“on PC of KDD, SIGIR, … and…”
Bootstrapping as graph proximity

NIPS

“...at NIPS, AISTATS, KDD and other learning conferences...”

SNOWBIRD

“For skiers, NIPS, SNOWBIRD, ... and...”

AISTATS

KDD

... “on PC of KDD, SIGIR, ... and...”

SIGIR

“... AISTATS,KDD,...”

shorter paths ~ earlier iterations
many paths ~ additional evidence
Similarity of Nodes in Graphs: Personal PageRank/RandomWalk with Restart

- Similarity defined by PageRank
- Similarity between nodes $x$ and $y$:
  
  “Random surfer model”: from a node $z$,
  
  with probability $\alpha$, stop and “output” $z$
  pick an edge label (rel) $r$ using $Pr(r \mid z)$ ...
  e.g. uniform
  pick a $y$ given $x$, $r$: e.g. uniform from $\{ y' : z \to y$ with label $r \}$
  
  repeat from node $y$ ....

  Similarity $x \sim y = Pr( \text{“output” } y \mid \text{start at } x)$

Bootstrapping: propagate from labeled data to “similar” unlabeled data.

Intuitively, $x \sim y$ is summation of weight of all paths from $x$ to $y$, where *weight* of path decreases exponentially with length.
PPR/RWR on a Graph

“William W. Cohen, CMU”

“Dr. W. W. Cohen”

dr

william

w

cmu

“Christos Faloutsos, CMU”

“George H. W. Bush”

“George W. Bush”
A little math exercise…

Let $x$ be less than 1 and larger than 0. Then

$$y = 1 + x + x^2 + x^3 + ... + x^n$$

$$y \approx (1 - x)^{-1}$$

Example: $x=0.1$, and $1+0.1+0.01+0.001+... = 1.11111 = 10/9$. 
Graph = Matrix

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>J</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

Graph representation:

- Nodes: A, B, C, D, E, F, G, H, I, J

The graph is represented as a matrix, with 1s indicating connections. For example, A-B indicates a connection between nodes A and B.
Graph = Matrix

Transitively Closed Components = “Blocks”

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>_</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>_</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>1</td>
<td>_</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td>_</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td>1</td>
<td>_</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
<td>_</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td></td>
<td>_</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
<td></td>
<td>_</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>_</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>_</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Of course we can’t see the “blocks” unless the nodes are sorted by cluster…
Graph = Matrix
Vector = Node → Weight

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>_</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>_</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>1</td>
<td>_</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td>_</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td>1</td>
<td>_</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>1</td>
<td></td>
<td>1</td>
<td>1</td>
<td>_</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td></td>
<td>_</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td></td>
<td></td>
<td>_</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>_</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>J</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>_</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Graph = Matrix

$M^*v_1 = v_2$ “propagates weights from neighbors”

$M \times v_1 = v_2$
A little math...

Let $W[i,j]$ be $\Pr(\text{walk to } j \text{ from } i)$ and let $\alpha$ be less than 1. Then:

\[
Y = I + \alpha W + (\alpha W)^2 + (\alpha W)^3 + ... (\alpha W)^n
\]

\[
Y(I - \alpha W) = (I + \alpha W + (\alpha W)^2 + (\alpha W)^3 + ...)(I - W)
\]

\[
Y(I - \alpha W) = (I - \alpha W) + (\alpha W - (\alpha W)^2 + ...)(I - W)
\]

\[
Y(I - \alpha W) = I - (\alpha W)^{n+1}
\]

\[
Y \approx (I - \alpha W)^{-1} \quad Y[i, j] = \frac{1}{Z} \Pr(j \mid i)
\]

The matrix $(I - \alpha W)$ is the Laplacian of $\alpha W$.

Generally the Laplacian is $(D - A)$ where $D[i,i]$ is the degree of $i$ in the adjacency matrix $A$. 
A little math...

Let $W[i,j]$ be $\Pr(\text{walk to } j \text{ from } i)$ and let $\alpha$ be less than 1. Then:

$$v^0 = \langle 0, 0, 0, \ldots, 0, 1, 0, \ldots, 0 \rangle$$

$$v^{t+1} = (1 - \alpha)v^0 + \alpha Wv^{t-1}$$

$$v^n \rightarrow Yv^0 \text{ so } v^n[j] \approx \Pr(j | i)$$

The matrix $(I - \alpha W)$ is the Laplacian of $\alpha W$.

Generally the Laplacian is $(D - A)$ where $D[i,i]$ is the degree of $i$ in the adjacency matrix $A$. 

Bootstrapping via PPR/RWR on graph of patterns and nodes

Key Ideas: So Far

• High-precision low-coverage extractors and large redundant corpora (macro-reading)
• Self-training/bootstrapping or co-training
• Other semi-supervised methods:
  - Expectation-maximization
  - Transductive margin-based methods (e.g., transductive SVM)
  - Graph-based methods
  - Label propagation via random walk with reset
Bootstrapping

Hearst ‘92

**Deeper linguistic features, free text…**

BlumMitchell ’98

**Learning, semi-supervised learning, dual feature spaces…**

Brin’ 98

**Scalability, surface patterns, use of web crawlers…**

Lin & Pantel ‘02

**Clustering by distributional similarity…**
Bootstrapping

Clustering by distributional similarity...

Hearst ‘92

Deeper linguistic features, free text...

Collins & Singer ‘99

Boosting-based co-train method using content & context features; context based on Collins’ parser; learn to classify three types of NE

BM’98

Learning, semi-supervised learning, dual feature spaces...

Brin’98

Scalability, surface patterns, use of web crawlers...
Bootstrapping

- **Hearst ‘92**
  - Deeper linguistic features, free text...
  - Lin & Pantel ‘02

- **BM’98**
  - Learning, semi-supervised learning, dual feature spaces...
  - Collins & Singer ‘99

- **Brin’98**
  - Scalability, surface patterns, use of web crawlers...
  - Riloff & Jones ‘99

- **Hearst-like patterns, Brin-like bootstrapping (+ “meta-level” bootstrapping) on MUC data**

Clustering by distributional similarity...
Bootstrapping

**Hearst ‘92**
- **Lin & Pantel ‘02**
  - Deeper linguistic features, free text...
- **Riloff & Jones ‘99**
- **Collins & Singer ‘99**

**BM’ 98**
- **Learning, semi-supervised learning, dual feature spaces**
  - EM like co-train method with context & content both defined by character-level tries

**Brin’ 98**
- **Scalability, surface patterns, use of web crawlers**
Bootstrapping

- Hearst ‘92
  - Lin & Pantel ‘02
  - Riloff & Jones ‘99
  - Collins & Singer ‘99

- BM’ 98
  - Deeper linguistic features, free text...
  - Learning, semi-supervised learning, dual feature spaces...
  - Cucerzan & Yarowsky ‘99 (morphology)

- Brin’ 98
  - Scalability, surface patterns, use of web crawlers...

- Clustering by distributional similarity...

Etzioni et al 2005
Bootstrapping

Clustering by distributional similarity...

Deeper linguistic features, free text...

Learning, semi-supervised learning, dual feature spaces...

Scalability, surface patterns, use of web crawlers...

Hearst ‘92

Lin & Pantel ‘02

Riloff & Jones ‘99

Collins & Singer ‘99

BM’ 98

Etzioni et al 2005

TextRunner

Brin’ 98

Cucerzan & Yarowsky ‘99

...
Bootstrapping

Hearst ‘92

Lin & Pantel ‘02

Deeper linguistic features, free text...

Riloff & Jones ‘99

Collins & Singer ‘99

BM ‘98

Learning, semi-supervised learning, dual feature spaces...

Cucerzan & Yarowsky ‘99

Etzioni et al 2005

Brin ‘98

Scalability, surface patterns, use of web crawlers...

NEll

TextRunner
OpenIE Demo

http://knowitall.github.io/openie/
Never Ending Language Learning

PI: Tom M. Mitchell

Machine Learning Department
Carnegie Mellon University
NELL Theses

1. we’ll never understand learning until we build never-ending machine learners

2. background knowledge is key to deep semantic analysis
   
   NELL KB, plus
   
   large scale corpus statistics
NELL today

Running 24x7, since January, 12, 2010

Today:
• knowledge base with ~100 million confidence-weighted beliefs
• learning to read
• gradually improving reading accuracy
• learning to reason
  gradually improving KB size,
> 100,000 learned rules, scalable probabilistic inference
• extending ontology
new relations: clustering typed pairs
new categories: (developing) clustering + reading subsets
  • beginning to include image analysis (via NEIL)
<table>
<thead>
<tr>
<th>instance</th>
<th>iteration</th>
<th>date</th>
<th>learned</th>
<th>confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>african americans at siege of petersburg 1 is a military conflict</td>
<td>938</td>
<td>10-jul-2015</td>
<td>90.6</td>
<td></td>
</tr>
<tr>
<td>david koch is a professor</td>
<td>934</td>
<td>25-jun-2015</td>
<td>100.0</td>
<td></td>
</tr>
<tr>
<td>california sacrament farm is a farm</td>
<td>934</td>
<td>25-jun-2015</td>
<td>99.0</td>
<td></td>
</tr>
<tr>
<td>estate referral services is a profession</td>
<td>934</td>
<td>25-jun-2015</td>
<td>94.4</td>
<td></td>
</tr>
<tr>
<td>japanese chicken wings is a type of meat</td>
<td>937</td>
<td>07-jul-2015</td>
<td>99.4</td>
<td></td>
</tr>
<tr>
<td>banc of america securities is a company in the economic sector of</td>
<td>934</td>
<td>25-jun-2015</td>
<td>99.6</td>
<td></td>
</tr>
<tr>
<td>investment</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fcc is headquartered in the city washington d c</td>
<td>939</td>
<td>16-jul-2015</td>
<td>96.9</td>
<td></td>
</tr>
<tr>
<td>patrick vieira plays for the team france</td>
<td>939</td>
<td>16-jul-2015</td>
<td>93.8</td>
<td></td>
</tr>
<tr>
<td>tom anderson is a top member of myspace</td>
<td>939</td>
<td>16-jul-2015</td>
<td>93.8</td>
<td></td>
</tr>
<tr>
<td>office is a synonym for united states department</td>
<td>934</td>
<td>25-jun-2015</td>
<td>100.0</td>
<td></td>
</tr>
</tbody>
</table>
NELL Is Improving Over Time (Jan 2010 to Nov 2014)

- **Number of NELL Beliefs vs. Time**
  - All beliefs: A plot showing a significant increase in the number of beliefs over time, starting from a few millions and increasing to billions.
  - High confidence beliefs: A plot showing a more gradual increase, starting from millions and increasing to tens of millions.

- **Reading Accuracy vs. Time**
  (Average over 31 predicates)
  - Precision@10: A plot showing an increase in precision from approximately 0.8 to 0.95 over the iterations.
  - Mean average precision: A plot showing an increase in mean average precision from approximately 0.3 to 0.45 over the iterations.

- **Human Feedback vs. Time**
  (Average 2.4 feedbacks per predicate per month)
  - A bar chart showing the distribution of human feedbacks, with the highest feedbacks for the lower predicate counts and a decrease as the predicate count increases.

[Mitchell et al., 2015]
Portuguese NELL

Recently-Learned

instance

adriane_galisteu is a...
basf_e_faber_caster is a...
manaus_cavaliers is a...
jacutinga_campina is a...
fim_da_guerra is a...
bamerindus is a...
nissan is a company...
susana_vieira is a...
campeonato_brasil is a...
toyota_mitsubishi is a...

conflitomilitar
(category)

See learned instances of conflitomilitar as a list or on a map.

Metadata

• allLearnedPatterns

"a armada durante _" "a causa diplomática _" "a armamentista durante _" "a declaração de capital...

Portuguese NELL
If: \( x_1 \) competes with \((x_1, x_2)\)

Then: economic sector \((x_1, x_3)\)
Inference by Random Walks

PRA: [Lao, Mitchell, Cohen, EMNLP 2011]

1. restrict precondition to a chain.
2. inference by random walks
3. combine multiple rule matches with log-linear model

If: \( x_1 \) competes with \((x_1, x_2)\)

Then: economic sector \((x_1, x_3)\)
Course Outline

1. Basic theories and practices on named entity recognition.

2. Recent advances in relation extraction:
   a. distant supervision
   b. latent variable models

3. Scalable IE and reasoning with first-order logics.
Recent Advances in IE: Distant Supervision
Relation Extraction

Predict relations between entities based on mentions (Cullota and Sorenson, 2004)

Example: learn the \textit{mascot} (\textit{object}, \textit{org}) relation.

Training data:

“A \textit{Scottish Terrier} has clearly won the hearts of the campus community and will become \textit{Carnegie Mellon’s new official mascot}”
Challenge

It is very expensive to obtain labeled training data.
Distant Supervision

Idea: if we know the relation between two entities, then any sentence that includes these two entities is likely to express the same relation.
Distant Supervision


Use a knowledge base of known relations to collect a lot of noisy training data.
Distant Supervision

Example: \text{mascot}(\text{Stanford\_tree},\text{Stanford\_Band}).

High quality examples:
“The \textit{Stanford Tree} is the \textit{Stanford Band}'s mascot.”
“Called — appropriately — the \textit{Stanford Tree}, it is the official mascot of the \textit{band}.”

Noisy examples:
“The \textit{Stanford band} invites you to be \textit{Tree} for a day.”
Distant Supervision: Pros

• **Has the advantages of supervised learning**
  o leverage rich, reliable hand-created knowledge
  o can use rich features (e.g. syntactic features)

• **Has the advantages of unsupervised learning**
  o leverage unlimited amounts of text data
  o allows for very large number of weak features
  o not sensitive to training corpus: genre independent
Mintz et al., (2009) ACL

Mintz, Bills, Snow, Jurafsky (2009).
Distant supervision for relation extraction without labeled data.

Training set
Freebase
102 relations
940,000 entities
1.8 million instances

Corpus
Wikipedia
1.8 million articles
25.7 million sentences
Frequent Freebase Relations

<table>
<thead>
<tr>
<th>Relation name</th>
<th>Size</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>/people/person/nationality</td>
<td>281,107</td>
<td>John Dugard, South Africa</td>
</tr>
<tr>
<td>/location/location/contains</td>
<td>253,223</td>
<td>Belgium, Nijlen</td>
</tr>
<tr>
<td>/people/person/profession</td>
<td>208,888</td>
<td>Dusa McDuff, Mathematician</td>
</tr>
<tr>
<td>/people/person/place_of_birth</td>
<td>105,799</td>
<td>Edwin Hubble, Marshfield</td>
</tr>
<tr>
<td>/dining/restaurant/cuisine</td>
<td>86,213</td>
<td>MacAyo’s Mexican Kitchen, Mexican</td>
</tr>
<tr>
<td>/business/business_chain/location</td>
<td>66,529</td>
<td>Apple Inc., Apple Inc., South Park, NC</td>
</tr>
<tr>
<td>/biology/organism_classification_rank</td>
<td>42,806</td>
<td>Scorpaeniformes, Order</td>
</tr>
<tr>
<td>/film/film/genre</td>
<td>40,658</td>
<td>Where the Sidewalk Ends, Film noir</td>
</tr>
<tr>
<td>/film/film/language</td>
<td>31,103</td>
<td>Enter the Phoenix, Cantonese</td>
</tr>
<tr>
<td>/biology/organism_higher_classification</td>
<td>30,052</td>
<td>Calopteryx, Calopterygidae</td>
</tr>
<tr>
<td>/film/film/country</td>
<td>27,217</td>
<td>Turtle Diary, United States</td>
</tr>
<tr>
<td>/film/writer/film</td>
<td>23,856</td>
<td>Irving Shulman, Rebel Without a Cause</td>
</tr>
<tr>
<td>/film/director/film</td>
<td>23,539</td>
<td>Michael Mann, Collateral</td>
</tr>
<tr>
<td>/film/producer/film</td>
<td>22,079</td>
<td>Diane Eskenazi, Aladdin</td>
</tr>
<tr>
<td>/people/deceased_person/place_of_death</td>
<td>18,814</td>
<td>John W. Kern, Asheville</td>
</tr>
<tr>
<td>/music/artist/origin</td>
<td>18,619</td>
<td>The Octopus Project, Austin</td>
</tr>
<tr>
<td>/people/person/religion</td>
<td>17,582</td>
<td>Joseph Chartrand, Catholicism</td>
</tr>
<tr>
<td>/book/author/works_written</td>
<td>17,278</td>
<td>Paul Auster, Travels in the Scriptorium</td>
</tr>
<tr>
<td>/soccer/football_position/players</td>
<td>17,244</td>
<td>Midfielder, Chen Tao</td>
</tr>
<tr>
<td>/people/deceased_person/cause_of_death</td>
<td>16,709</td>
<td>Richard Daintree, Tuberculosis</td>
</tr>
<tr>
<td>/film/film/music</td>
<td>14,070</td>
<td>Stavisky, Stephen Sondheim</td>
</tr>
<tr>
<td>/business/company/industry</td>
<td>13,805</td>
<td>ATS Medical, Health care</td>
</tr>
</tbody>
</table>
Collecting Training Data

Corpus text

- Bill Gates founded Microsoft in 1975.
- Bill Gates, founder of Microsoft, ...
- Bill Gates attended Harvard from ...
- Google was founded by Larry Page ...

Training data

Freebase

- Founder: (Bill Gates, Microsoft)
- Founder: (Larry Page, Google)
- CollegeAttended: (Bill Gates, Harvard)
Collecting Training Data

Corpus text

- Bill Gates founded Microsoft in 1975.
- Bill Gates, founder of Microsoft, ...
- Bill Gates attended Harvard from...
- Google was founded by Larry Page ...

Training data

- (Bill Gates, Microsoft) 
  - Label: Founder
  - Feature: X founded Y

Freebase

- Founder: (Bill Gates, Microsoft)
- Founder: (Larry Page, Google)
- CollegeAttended: (Bill Gates, Harvard)
Collecting Training Data

Corpus text

Bill Gates founded Microsoft in 1975. Bill Gates, founder of Microsoft, …
Bill Gates attended Harvard from…
Google was founded by Larry Page …

Training data

(Bill Gates, Microsoft)
Label: Founder
Feature: X founded Y
Feature: X, founder of Y

Freebase

Founder: (Bill Gates, Microsoft)
Founder: (Larry Page, Google)
CollegeAttended: (Bill Gates, Harvard)
Processing Testing Data

Corpus text

Henry Ford founded Ford Motor Co. in…
Ford Motor Co. was founded by Henry Ford…
Steve Jobs attended Reed College from…

Test data

(Henry Ford, Ford Motor Co.)
Label: ???
Feature: X founded Y
Feature: Y was founded by X
The Experiment

Positive training data
- (Bill Gates, Microsoft)
  - Label: Founder
  - Feature: X
  - founded Y
- (Bill Gates, Harvard)
  - Label: CogArchfieldName
  - Feature: X
  - attended Y
- (Larry Page, Google)
  - Label: Founder
  - Feature: Y was
  - founded by X

Negative training data
- (Larry Page, Microsoft)
  - Label: NOTREADfieldName
  - X took a
  - swipe at Y
- (Larry Page, Harvard)
  - Label: NOTREADfieldName
  - X
- (Bill Gates, Google)
  - Label: NOTREADfieldName
  - Y invited
  - worst fear

Learning: multiclass logistic regression

Test data
- (Henry Ford, Ford Motor Co.)
  - Label: ???
  - Feature: X
  - founded Y
  - founded by X
- (Steve Jobs, Reed College)
  - Label: ???
  - Feature: X
  - attended Y

Trained relation classifier

Predictions!
- (Henry Ford, Ford Motor Co.)
  - Label: Founder
- (Steve Jobs, Reed College)
  - Label: CollegeAttended
Lexical and Dependency Path Features

Astronomer Edwin Hubble was born in Marshfield, Missouri.
Experimental Settings

• 1.8 million relation instances used for training

• 800,000 Wikipedia articles used for training, 400,000 different articles used for testing

• Only extract relation instances not already in Freebase
Learned Relational Facts

<table>
<thead>
<tr>
<th>Relation name</th>
<th>New instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>/location/location/contains</td>
<td>Paris, Montmartre</td>
</tr>
<tr>
<td>/location/location/contains</td>
<td>Ontario, Fort Erie</td>
</tr>
<tr>
<td>/music/artist/origin</td>
<td>Mighty Wagon, Cincinnati</td>
</tr>
<tr>
<td>/people/deceased_person/place_of_death</td>
<td>Fyodor Kamensky, Clearwater</td>
</tr>
<tr>
<td>/people/person/nationality</td>
<td>Marianne Yvonne Heemskerk, Netherlands</td>
</tr>
<tr>
<td>/people/person/place_of_birth</td>
<td>Wavell Wayne Hinds, Kingston</td>
</tr>
<tr>
<td>/book/author/works_written</td>
<td>Upton Sinclair, Lanny Budd</td>
</tr>
<tr>
<td>/business/company/founders</td>
<td>WWE, Vince McMahon</td>
</tr>
<tr>
<td>/people/person/profession</td>
<td>Thomas Mellon, judge</td>
</tr>
</tbody>
</table>
# Human Evaluation

Precision, using Mechanical Turk labelers:

<table>
<thead>
<tr>
<th>Relation name</th>
<th>100 instances</th>
<th></th>
<th></th>
<th>1000 instances</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Syn</td>
<td>Lex</td>
<td>Both</td>
<td>Syn</td>
<td>Lex</td>
<td>Both</td>
</tr>
<tr>
<td>/film/director/film</td>
<td>0.49</td>
<td>0.43</td>
<td>0.44</td>
<td>0.49</td>
<td>0.41</td>
<td>0.46</td>
</tr>
<tr>
<td>/film/writer/film</td>
<td>0.70</td>
<td>0.60</td>
<td>0.65</td>
<td>0.71</td>
<td>0.61</td>
<td>0.69</td>
</tr>
<tr>
<td>/geography/river/basin_countries</td>
<td>0.65</td>
<td>0.64</td>
<td>0.67</td>
<td>0.73</td>
<td>0.71</td>
<td>0.64</td>
</tr>
<tr>
<td>/location/country/administrative_divisions</td>
<td>0.68</td>
<td>0.59</td>
<td>0.70</td>
<td>0.72</td>
<td>0.68</td>
<td>0.72</td>
</tr>
<tr>
<td>/location/location/contains</td>
<td>0.81</td>
<td>0.89</td>
<td>0.84</td>
<td>0.85</td>
<td>0.83</td>
<td>0.84</td>
</tr>
<tr>
<td>/location/us_county/country_seat</td>
<td>0.51</td>
<td>0.51</td>
<td>0.53</td>
<td>0.47</td>
<td>0.57</td>
<td>0.42</td>
</tr>
<tr>
<td>/music/artist/origin</td>
<td>0.64</td>
<td>0.66</td>
<td>0.71</td>
<td>0.61</td>
<td>0.63</td>
<td>0.60</td>
</tr>
<tr>
<td>/people/deceased_person/place_of_death</td>
<td>0.80</td>
<td>0.79</td>
<td>0.81</td>
<td>0.80</td>
<td>0.81</td>
<td>0.78</td>
</tr>
<tr>
<td>/people/person/nationality</td>
<td>0.61</td>
<td>0.70</td>
<td>0.72</td>
<td>0.56</td>
<td>0.61</td>
<td>0.63</td>
</tr>
<tr>
<td>/people/person/place_of_birth</td>
<td>0.78</td>
<td>0.77</td>
<td>0.78</td>
<td>0.88</td>
<td>0.85</td>
<td>0.91</td>
</tr>
<tr>
<td>Average</td>
<td>0.67</td>
<td>0.66</td>
<td>0.69</td>
<td>0.68</td>
<td>0.67</td>
<td>0.67</td>
</tr>
</tbody>
</table>
Steve Jobs presents Apple’s HQ.
Apple CEO Steve Jobs ...
Steve Jobs holds Apple stock.
Steve Jobs, CEO of Apple, ...
Google’s takeover of Youtube ...
Youtube, now part of Google, ...
Apple and IBM are public.
... Microsoft’s purchase of Skype.

CEO-of(1,2)
N/A(1,2)
Acquired(1,2)
Acquired(1,2)
CEO-of(Rob Iger, Disney)
CEO-of(Steve Jobs, Apple)
Acquired(Google, Youtube)
Acquired(Msft, Skype)
Acquired(Citigroup, EMI)
Mintz et al. (2009)

Issues?

• No multi-instance learning

• No multi-relation learning
Multi-Instance Learning

Steve Jobs presents Apple’s HQ.  
Apple CEO Steve Jobs ...  
Steve Jobs holds Apple stock.  
Steve Jobs, CEO of Apple, ...  
Google’s takeover of Youtube ...  
Youtube, now part of Google, ...  
Apple and IBM are public.  
... Microsoft’s purchase of Skype.  

Cf. [Bunescu, Mooney 07], [Riedel, Yao, McCallum 10]

CEO-of(Rob Iger, Disney)  
CEO-of(Steve Jobs, Apple)  
Acquired(Google, Youtube)  
Acquired(Msft, Skype)  
Acquired(Citigroup, EMI)
Overlapping Relations

Steve Jobs presents Apple’s HQ.

Apple CEO Steve Jobs ...

Steve Jobs holds Apple stock.

Steve Jobs, CEO of Apple, ...

Google’s takeover of Youtube ...

Youtube, now part of Google, ...

Apple and IBM are public.

... Microsoft’s purchase of Skype ...

SH-of(Steve Jobs, Apple)
CEO-of(Rob Iger, Disney)
CEO-of(Steve Jobs, Apple)
Acquired(Google, Youtube)
Acquired(Msft, Skype)
Acquired(Citigroup, EMi)
Hoffman et al. (2011)

Knowledge-Based Weak Supervision for Information Extraction of Overlapping Relations

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, Daniel S. Weld
Computer Science & Engineering
University of Washington
Seattle, WA 98195, USA
{raphaelh, clzhang, xiaoling, lsz, weld}@cs.washington.edu
Sentence-Level Learning

Steve Jobs presents Apple’s HQ.

Apple CEO Steve Jobs ...

Steve Jobs holds Apple stock.

Steve Jobs, CEO of Apple, ...

Google’s takeover of Youtube ...

Youtube, now part of Google, ...

Apple and IBM are public.

... Microsoft’s purchase of Skype.

Train so that extracted facts match facts in DB

CEO-of(Rob Iger, Disney)
CEO-of(Steve Jobs, Apple)
Acquired(Google, Youtube)
Acquired(Msft, Skype)
Acquired(Citigroup, EMI)
Steve Jobs, Apple:

Steve Jobs was founder of Apple. Steve Jobs, Steve Wozniak and Ronald Wayne founded Apple. Steve Jobs is CEO of Apple.

$$p(Y = y, Z = z | x; \theta) \overset{\text{def}}{=} \frac{1}{Z_x} \prod_{r} \Phi_{\text{join}}(y^r, z) \prod_{i} \Phi_{\text{extract}}(z_i, x_i)$$

$$\Phi_{\text{join}}(y^r, z) \overset{\text{def}}{=} \begin{cases} 1 & \text{if } y^r = \text{true} \land \exists i: z_i = r \\ 0 & \text{otherwise} \end{cases}$$

All features at sentence-level

(joint factors are deterministic ORs)
Inference

Computing  \[ \arg \max_z p(z|x, y; \theta) : \]

Steve Jobs was founder of Apple.

Steve Jobs, Steve Wozniak and Ronald Wayne founded Apple.

Steve Jobs is CEO of Apple.
Inference

Variant of the weighted, edge-cover problem:

Steve Jobs was founder of Apple.  
Steve Jobs, Steve Wozniak and Ronald Wayne founded Apple.  
Steve Jobs is CEO of Apple.
Learning

Training set \( \{(x_i, y_i)|i = 1 \ldots n\} \), where

- \( i \) corresponds to a particular entity pair
- \( x_i \) contains all sentences with mentions of pair
- \( y_i \) bit vector of facts about pair from database

Maximize Likelihood

\[
O(\theta) = \prod_i p(y_i|x_i; \theta) = \prod_i \sum_z p(y_i, z|x_i; \theta)
\]
Sentential vs. Aggregate Extraction

Sentential

Input: one sentence

Steve Jobs is CEO of Apple, ...

Aggregate

Input: one entity pair

<Steve Jobs, Apple>

Steve Jobs was founder of Apple.
Steve Jobs, Steve Wozniak and Ronald Wayne founded Apple.
Steve Jobs is CEO of Apple.

CEO-of(1,2)
Distant Supervision: Related Work

- Mintz, Bills, Snow, Jurafsky 09:
  Extraction at aggregate level
  Features: conjunctions of lexical, syntactic, and entity type info along dependency path

- Riedel, Yao, McCallum 10:
  Extraction at aggregate level
  Latent variable on sentence

- Bunescu, Mooney 07:
  Multi-instance learning for relation extraction
  Kernel-based approach
Experimental Setup

• Data as in Riedel et al. 10:
  LDC NYT corpus, 2005-06 (training), 2007 (testing)
  Data first tagged with Stanford NER system
  Entities matched to Freebase, ~ top 50 relations
  Mention-level features as in Mintz et al. 09

• Systems:
  MultiR: proposed approach
  SoloR: re-implementation of Riedel et al. 2010
7.2 Sentential Extraction

Although their model includes variables to model sentential extraction, Riedel et al. did not report sentence level performance. To generate the precision vs. recall curve we used the joint model as assignment score for each of the sentences that contributed to the aggregate extraction decision.

Figure 1 shows approximate precision vs. recall curves for MULTIR and SOLOR computed against manually generated sentence labels as defined in Section 6. MULTIR achieves significantly higher recall with a consistently high level of precision. At the highest recall point, MULTIR reaches (0.6, x precision and 0.6% recall, for an F0 score of 0.96/x).

7.3 Relation-Specific Performance

Since the data contains an unbalanced number of instances of each relation, we also report precision and recall for each of the ten most frequent relations. Let $S_{M}$ be the sentences where MULTIR extracted an instance of relation $r$, and let $S_{F}$ be the sentences that match the arguments of a fact about relation $r$. For each $r$, we sample 0.99 sentences from both $S_{M}$ and $S_{F}$ and manually check accuracy.

To estimate precision $\tilde{P}_r$, we compute the ratio of true relation mentions in $S_{M}$ and to estimate recall $\tilde{R}_r$, we take the ratio of true relation mentions in $S_{F}$ which are returned by our system.

Table 0 presents this approximate precision and recall for MULTIR on each of the relations, along with statistics we computed to measure the quality of the weak supervision. Precision is high for the majority of relations but recall is consistently lower. We also see that the Freebase matches are highly skewed in quantity and can be low quality for some relations, with very few of them actually corresponding to true extractions. The approach generally performs best on the relations with a sufficiently large number of true matches, in many cases even achieving precision that outperforms the accuracy of the heuristic matches at reasonable recall levels.

7.4 Overlapping Relations

Table 0 also highlights some of the effects of learning with overlapping relations. For example, in the data, almost all of the matches for the administrative divisions relation overlap with the contains relation, because they both model relationships for a pair of locations. Since, in general, sentences are much more likely to describe a contains relation, this overlap leads to a situation were almost none of the administrative division matches are true ones, and we cannot accurately learn an extractor. However, we can still learn to accurately extract the contains relation, despite the distracting matches. Similarly, the place of birth and place of death relations tend to overlap, since it is often the case that people are born and die in the same city. In both cases, the precision outperforms the labeling accuracy and the recall is relatively high.

To measure the impact of modeling overlapping relations, we also evaluated a simple, restricted baseline. Instead of labeling each entity pair with the set of all true Freebase facts, we created a dataset where each true relation was used to create a different training example. Training MULTIR on this data simulates effects of conflicting supervision that can come from not modeling overlaps. On average across relations, precision increases 0. points but recall drops 0. points, for an overall reduction in F0 score from 0.96/x to 0.96/x.
Distant Supervision: Conclusion

• Widely used in the IE community nowadays.
• A much cheaper way of obtaining training data
• Still, there’s room for improvement:
  • what about entities that are not in Freebase?
  • what if entities are in Freebase, but no relation is recorded?
Recent Advances in IE: Latent Variable Modeling
Universal Schema

• Riedel et al., NAACL 2013. Relation Extraction with Matrix Factorization and Universal Schemas.

• Motivation: use matrix representation for relation extraction.

• Idea: put all training and testing data into a matrix, and fill in the missing values.

• Jointly learn latent factor representation for surface patterns and multiple relations.
### Universal Schema

- **Rows:** pair of entities. E.g., (William, CMU)
- **Columns:** surface patterns and relations. E.g., X-is_a_professor_at-Y teaches (X, Y)
Matrix Factorization

- Approach: Bayesian Personalized Ranking (Rendle et al., 2009)
- Requires: negative training data.
- How to collect negative data: both entities of the entity pair occur in Freebase, however, Freebase does not say there is a relation between them.
Performance

Universal Schema

• Pros:
  1) language, schema independent
  2) joint learning of surface patterns and relations
  3) scalability

• Cons:
  1) explainability
  2) requires negative examples
Course Outline

1. Basic theories and practices on named entity recognition: supervised and semi-supervised.

2. Recent advances in relation extraction:
   a. distant supervision
   b. latent variable models

3. Scalable IE and reasoning with first-order logics.
Joint IE and Reasoning
A Motivating Example…

An elementary school student was sent to detention by his Math teacher after school. When he got home, his father said: “Ma Yun, what happen to you at school today?” Ma: “Sorry dad, I was playing with a magnet, but it attracted Mrs. Smith’s golden ring. Then, Mrs. Smith went out to cry, and slapped the P.E. teacher in the face.”

Query:
Who is most likely the husband of Mrs. Smith?

This example was adapted from Weibo.
An elementary school student was sent to detention by his Math teacher after school. When he got home, his father said: “Ma Yun, what happen to you at school today?” : “Sorry dad, I was playing with a magnet, but it attracted Mrs. Smith’s golden ring. Then, Mrs. Smith went out to cry, and slapped the P.E. teacher in the face.”

This example was adapted from Weibo.
Issues with Modern IE Systems

- No relational KB inference is performed at extraction time (or no inference at all).
- Classification is not the panacea.
- Big pipeline: error cascades.
Motivations

• To deal with complexity, we need first-order logics to perform reasoning.

• To deal with uncertainty, we need statistical/probabilistic approaches, at the same time.
Issues with KB Reasoning Systems

• Often done using relational triples (e.g., wife(barack,michelle)) after IE, and key contextual information is lost.

E.g., Path-Ranking Algorithm (Ni et al., 2010)

*PRA Paths for inferring athletePlaysSport:*

athletePlaysSport(A,S):- factAthletePlaysForTeam(A,T),factTeamPlaysSport(T,S).

*PRA Paths for inferring teamPlaysSport:*

teamPlaysSport(T,S):-
    factMemberOfConference(T,C),factConferenceHasMember(C,T'),factTeamPlaysSport(T',S).

teamPlaysSport(T,S):-
    factTeamHasAthlete(T,A),factAthletePlaysSport(A,S).
Our Approach

• presents a joint IE and reasoning model in a statistical relational learning setting;
• incorporates latent contexts into probabilistic first-order logics.
Agenda

• Motivation
• Background: ProPPR
• Datasets
• Joint IE and Structure Learning
• Experiments
• Conclusion
Wait, Why Not Markov Logic Network?

network size is $O(n^a)$, where $a = \#\text{arity}$.

e.g., \texttt{holdStock(person,company)}

Inference time often depends on graph size.
Programming with Personalized PageRank (ProPPR)

- CIKM 2013 best paper honorable mention
- is a probabilistic first-order logic
- can be used in:
  - entity resolution, classification (Wang et al., 2013)
  - dependency parsing (Wang et al., 2014 EMNLP)
  - large-scale KB inference (Wang et al., 2015 MLJ)
  - logic programming (Wang et al., 2015 IJCAI)
Inference Time Comparison

ProPPR’s inference time is independent of the size of the graph (Wang et al., 2013).
Accuracy: Citation Matching

<table>
<thead>
<tr>
<th></th>
<th>Cites</th>
<th>Authors</th>
<th>Venues</th>
<th>Titles</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLN</td>
<td>0.513</td>
<td>0.532</td>
<td>0.602</td>
<td>0.544</td>
</tr>
<tr>
<td>ProPPR (w=1)</td>
<td>0.680</td>
<td>0.836</td>
<td>0.860</td>
<td>0.908</td>
</tr>
<tr>
<td>ProPPR</td>
<td>0.800</td>
<td>0.840</td>
<td>0.869</td>
<td>0.900</td>
</tr>
</tbody>
</table>

AUC scores: 0.0=low, 1.0=high

w=1 is before learning (i.e., heuristic matching rules, weighted with PPR)
ProPPR Example

Input:

Query: `about(a,?)`

```
Fashion

a: "Olympic sprinter..."
b: "Model Reeva..."
c: "Champion sprinter...
d: "Today..."
```

```
sport
```
An Example ProPPR Program

about(X,Z) :- handLabeled(X,Z) # base.
about(X,Z) :- sim(X,Y),about(Y,Z) # prop.
sim(X,Y) :- links(X,Y) # sim,link.
sim(X,Y) :-
    hasWord(X,W),hasWord(Y,W),
    linkedBy(X,Y,W) # sim,word.
linkedBy(X,Y,W) :- true # by(W).

Feature Vector

Feature Template
Program + DB + Query define a proof graph, where nodes are conjunctions of goals and edges are labeled with sets of features.
Transition probabilities, \( \Pr(\text{child}|\text{parent}) \), plus Personalized PageRank (aka Random-Walk-With-Reset) define a distribution over nodes.

Very fast approximate methods for PPR

Learning via pSGD
Approximate Inference in ProPPR

• Score for a query soln (e.g., “Z=sport” for “about(a,Z)”) depends on probability of reaching a ☐ node*

“Grounding” (proof tree) size is $O(1/\alpha \varepsilon)$ ... ie independent of DB size $\Rightarrow$ fast approx incremental inference (Reid,Lang,Chung, 08)

---

$\alpha$ is reset probability

*as in Stochastic Logic Programs [Cussens, 2001]

Basic idea: incrementally expand the tree from the query node until all nodes $v$ accessed have weight below $\varepsilon/\text{degree}(v)$
Parameter Learning in ProPPR

PPR probabilities are a stationary distribution of a Markov chain

\[ p^{t+1} \equiv \alpha s + (1 - \alpha)Mp^t \]

Transition probabilities \( u \rightarrow v \) are derived by linearly combining features of an edge, applying a squashing function \( f \), and normalizing

\[
\begin{align*}
 s_{uv} & \equiv \vec{\phi}_{uv} \cdot \vec{w} \\
 t_u & \equiv \sum_{v'} f(s_{uv'}) \\
 M_{u,v} & \equiv \frac{f(s_{uv})}{t_u}
\end{align*}
\]

\( f \) is exp, truncated tanh, ReLU…
PPR probabilities are a stationary distribution of a Markov chain

\[ p^{t+1} = \alpha s + (1 - \alpha) M p^t \]

Learning uses gradient descent: derivative \( d^t \) of \( p^t \) is:

\[ d^{t+1} = \frac{\partial}{\partial w} p^{t+1} = (1 - \alpha) \left( \left( \frac{\partial}{\partial w} M \right) p^t + M d^t \right) \]

Overall algorithm not unlike backprop...we use parallel SGD
Where Does the Program Come From?

• Traditionally by hand.
• We use structure learning to automatically learn first-order logic clauses from data.
• Idea (CIKM 2014):
  build a second-order abductive logic
  whose parameters correspond to 1\textsuperscript{st}-order theory
  reduce the structure learning to parameter learning.
Logic program is an *interpreter* for a program containing *all possible rules* from a sublanguage.

DB₀: sister(malia,sasha), mother(malia,michelle), …

DB: rel(sister,malia,sasha), rel(mother,malia,michelle), …

**Interpreter** for all clauses of the form \( P(X,Y) \) :- \( Q(X,Y) \):

\[
\text{interp}(P,X,Y) :- \text{rel}(P,X,Y). \\
\text{interp}(P,X,Y) :- \text{interp}(Q,X,Y), \text{assumeRule}(P,Q). \\
\text{assumeRule}(P,Q) :- \text{true} \quad \# f(P,Q). \quad // P(X,Y) :- Q(X,Y)
\]

Query₀: sibling(malia,Z)
Query: interp(sibling, malia, Z)

Features correspond to *specific* rules:

\[
\text{assumeRule}(\text{sibling}, \text{sister}), \ldots \\
\]
\[
f(\text{sibling}, \text{sister}) \\
\]

\[
\ldots
\]

\[
Z=\text{sasha}
\]

\[
\text{assumeRule}(\text{sibling}, \text{mother}), \ldots \\
\]
\[
f(\text{sibling}, \text{mother}) \\
\]

\[
\ldots
\]

\[
Z=\text{michelle}
\]
Logic program is an *interpreter* for a program containing all possible rules from a sublanguage.

Features ~ rules. For example:
\[ f(\text{sibling}, \text{sister}) \approx \text{sibling}(X,Y):- \text{sister}(X,Y) \]

Gradient of parameters (feature weights) informs you about what *rules* could be added to the theory...

**Interpret**er for all clauses of the form \( P(X,Y) :- Q(X,Y) \):

\[
\begin{align*}
\text{interp}(P,X,Y) &:- \text{rel}(P,X,Y). \\
\text{interp}(P,X,Y) &:- \text{interp}(Q,X,Y), \text{assumeRule}(P,Q).
\end{align*}
\]

\( \text{assumeRule}(P,Q) :- \text{true} \quad \# f(P,Q). \quad \text{// } P(X,Y) :- Q(X,Y) \)

**Added rule:**
\[
\text{Interp}(\text{sibling},X,Y) :- \text{interp}(\text{sister},X,Y).
\]

\[ f(\text{sibling}, \text{sister}) \]

\[ Z = \text{sasha} \]

\[ Z = \text{michelle} \]
Joint IE and Structure learning
Data Collection

(1) DBpedia Outlinks

(2) Wikipedia Page

"Louis was born in Paris, the son of Philip I..."

(3) <a href="http://en.wikipedia.org/wiki/Philip_I_of_France>...

(4) DBpedia Infobox Relation Mapping

Entity: Louis VI of France

Entity: Philip I of France

Parent (Louis VI, Philip I)

(1) Dbpedia Supercategory: European royal families
# Joint IE+SL Theory

<table>
<thead>
<tr>
<th>Rule template</th>
<th>ProPPR clause</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Structure learning</strong></td>
<td></td>
</tr>
<tr>
<td>(a) ( P(X,Y) ) ( \leftarrow ) ( R(X,Y) )</td>
<td>interp( (P,X,Y) ) ( \leftarrow ) interp0( (R,X,Y) ), abduce_if( (P,R) ). abduce_if( (P,R) ) ( \leftarrow ) true # f_if( (P,R) ).</td>
</tr>
<tr>
<td>(b) ( P(X,Y) ) ( \leftarrow ) ( R(Y,X) )</td>
<td>interp( (P,X,Y) ) ( \leftarrow ) interp0( (R,Y,X) ), abduce_ifInv( (P,R) ). abduce_ifInv( (P,R) ) ( \leftarrow ) true # f_ifInv( (P,R) ).</td>
</tr>
<tr>
<td>(c) ( P(X,Y) ) ( \leftarrow ) ( R1(X,Z), R2(Z,Y) )</td>
<td>interp( (P,X,Y) ) ( \leftarrow ) interp0( (R1,X,Z) ), interp0( (R2,Z,Y) ), abduce_chain( (P,R1,R2) ). abduce_chain( (P,R1,R2) ) ( \leftarrow ) true # f_chain( (P,R1,R2) ).</td>
</tr>
<tr>
<td>base case for SL interpreter insertion point for learned rules</td>
<td>interp0( (P,X,Y) ) ( \leftarrow ) rel( (R,X,Y) ). interp0( (P,X,Y) ) ( \leftarrow ) any rules learned by SL.</td>
</tr>
<tr>
<td><strong>Information extraction</strong></td>
<td></td>
</tr>
<tr>
<td>(d) ( R(X,Y) ) ( \leftarrow ) ( \text{link}(X,Y,W) ), indicates( (W,R) ).</td>
<td>interp( (R,X,Y) ) ( \leftarrow ) ( \text{link}(X,Y,W) ), abduce_indicates( (W,R) ). abduce_indicates( (W,R) ) ( \leftarrow ) true # f_ind1( (W,R) ).</td>
</tr>
<tr>
<td>(e) ( R(X,Y) ) ( \leftarrow ) ( \text{link}(X,Y,W1), \text{link}(X,Y,W2), \text{indicates}(W1,W2,R) ).</td>
<td>interp( (R,X,Y) ) ( \leftarrow ) ( \text{link}(X,Y,W1), \text{link}(X,Y,W2) ), abduce_indicates( (W1,W2,R) ). abduce_indicates( (W1,W2,R) ) ( \leftarrow ) true # f_ind2( (W1,W2,R) ).</td>
</tr>
</tbody>
</table>
Experiments

• Task: KB Completion.
• Three Wikipedia Datasets:
  royal, geo, american.
  67K, 12K, and 43K links respectively.

<table>
<thead>
<tr>
<th></th>
<th>10% deleted</th>
<th>50% deleted</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProPPR/SL</td>
<td>79.5</td>
<td>61.9</td>
</tr>
<tr>
<td>ProPPR/IE</td>
<td>81.1</td>
<td>70.6</td>
</tr>
</tbody>
</table>

Results on Royal, similar results on two other InfoBox datasets.
Joint Relation Learning IE in ProPPR

• Experiment
  Combine IE and SL rules

<table>
<thead>
<tr>
<th></th>
<th>10% deleted</th>
<th>50% deleted</th>
</tr>
</thead>
<tbody>
<tr>
<td>ProPPR/SL</td>
<td>79.5</td>
<td>61.9</td>
</tr>
<tr>
<td>ProPPR/IE</td>
<td>81.1</td>
<td>70.6</td>
</tr>
<tr>
<td>ProPPR/Joint IE,SL</td>
<td>82.8</td>
<td>78.6</td>
</tr>
</tbody>
</table>

Similar results on two other InfoBox datasets
Joint IE and Relation Learning

- Baselines: MLNs (Richardson and Domingos, 2006), Universal Schema (Riedel et al., 2013), IE- and structure-learning-only models.

<table>
<thead>
<tr>
<th>% missing</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baselines</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MLN</td>
<td>60.8</td>
<td>43.7</td>
<td>44.9</td>
<td>38.8</td>
<td>38.8</td>
</tr>
<tr>
<td>Universal Schema</td>
<td>48.2</td>
<td>53.0</td>
<td>52.9</td>
<td>47.3</td>
<td>41.2</td>
</tr>
<tr>
<td>SL</td>
<td>79.5</td>
<td>77.2</td>
<td>74.8</td>
<td>65.5</td>
<td>61.9</td>
</tr>
<tr>
<td>IE only</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IE (U)</td>
<td>81.3</td>
<td>78.5</td>
<td>76.4</td>
<td>75.7</td>
<td>70.6</td>
</tr>
<tr>
<td>IE (U+B)</td>
<td>81.1</td>
<td>78.1</td>
<td>76.2</td>
<td>75.5</td>
<td>70.3</td>
</tr>
<tr>
<td>Joint</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SL+IE (U)</td>
<td>82.8</td>
<td>80.9</td>
<td>79.1</td>
<td>77.9</td>
<td>78.6</td>
</tr>
<tr>
<td>SL+IE (U+B)</td>
<td>83.4</td>
<td>82.0</td>
<td>80.7</td>
<td>79.7</td>
<td>80.3</td>
</tr>
</tbody>
</table>
Latent Context Invention

Making the classifier more powerful: introduce latent classes (analogous to invented predicates) which can be combined with the context words in the features used by the classifier.

\[
\begin{align*}
\text{Latent context invention} \\
(f) \quad R(X, Y) &\colon= \text{latent}(L), \\
&\quad \text{link}(X, Y, W), \\
&\quad \text{indicate}(W, L, R) \\
&\quad \text{interp}(R, X, Y) :\text{latent}(L), \text{link}(X, Y, W), \text{abduce}_{\text{latent}}(W, L, R). \\
&\quad \text{abduce}_{\text{latent}}(W, L, R) :\text{true} \#f_{\text{latent}1}(W, L, R). \\
\end{align*}
\]

\[
\begin{align*}
(g) \quad R(X, Y) &\colon= \text{latent}(L_1), \text{latent}(L_2) \\
&\quad \text{link}(X, Y, W), \\
&\quad \text{indicate}(W, L_1, L_2, R) \\
&\quad \text{interp}(R, X, Y) :\text{latent}(L_1), \text{latent}(L_2), \text{link}(X, Y, W), \\
&\quad \quad \quad \text{abduce}_{\text{latent}}(W, L_1, L_2, R). \\
&\quad \text{abduce}_{\text{latent}}(W, L_1, L_2, R) :\text{true} \#f_{\text{latent}2}(W, L_1, L_2, R). \\
\end{align*}
\]
Joint IE and Relation Learning

- Task: Knowledge Base Completion.
- Baselines: MLNs (Richardson and Domingos, 2006), Universal Schema (Riedel et al., 2013), IE- and structure-learning-only models.

![Table](image.png)
Explaining the Parameters

\[\text{indicates}(\text{“mother”}, \text{parent})\]
\[\text{indicates}(\text{“king”}, \text{parent})\]
\[\text{indicates}(\text{“spouse”}, \text{spouse})\]
\[\text{indicates}(\text{“married”}, \text{spouse})\]
\[\text{indicates}(\text{“succeeded”}, \text{successor})\]
\[\text{indicates}(\text{“son”}, \text{successor})\]

\[\text{parent}(X,Y) : \text{ successor}(Y,X)\]
\[\text{successor}(X,Y) : \text{ parent}(Y,X)\]
\[\text{spouse}(X,Y) : \text{ spouse}(Y,X)\]
\[\text{parent}(X,Y) : \text{ predecessor}(X,Y)\]
\[\text{successor}(Y,X) : \text{ spouse}(X,Y)\]
\[\text{predecessor}(X,Y) : \text{ parent}(X,Y)\]
Discussions

• Comparing to latent variable models, our method is explainable.
• This is multi-instance multi-relation distant supervision with logic.
• This framework allows us to recursively learn relations, and jointly reason with IE clauses.
• Our structure learning method is efficient: according to Kok & Domingos‘s (2010, ICML), LSM sometimes takes 28 days to learn on a moderate-small dataset, where as our method needs a few minutes on a similar-sized dataset.
Conclusion

• We introduce a probabilistic logic programming method for joint IE and reasoning.
• We briefly show how to incorporate latent classes in first-order logic.
• Our system outperforms state-of-the-art IE systems.
ProPPR Demo
Course Conclusion

1. Basic theories and practices on named entity recognition: supervised, semi-supervised, and unsupervised.

2. Recent advances in relation extraction:
   a. distant supervision
   b. latent variable models

3. Scalable IE and reasoning with first-order logics.
Acknowledgement

- CIPS Executives
- Peking University
- General Chair: Prof. Le Sun
- PC Chair: Prof. Heng Ji
- Volunteers
- Participants
Ask Me Anything!

yww@cs.cmu.edu