Learning 5000 Relational Extractors

Raphael Hoffmann
University of Washington

(joint work with Congle Zhang and Daniel S. Weld)

Talk at Affiliates Meeting
10/27/2010
“What Russian-born writers publish in the U.S.?”

Use Information Extraction
Aksakov was born in what is now Bashkortostan.

Garshin was born in Ekaterinoslav Province, Russia.

Figner was a Russian revolutionary and narodnik born June 25, 1852 in Kazan, Russia.

Dovlatov’s birthplace was Ufa, Republic of Bashkiria, USSR.
# Types of Information Extraction

**Input**

<table>
<thead>
<tr>
<th>Traditional, Supervised IE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corpus + Manual Labels</td>
</tr>
</tbody>
</table>

**Relations**

<table>
<thead>
<tr>
<th>Specified in Advance</th>
<th></th>
</tr>
</thead>
</table>
Aksakov was born in what is now Bashkortostan.
NNP VBD VBN WP VBZ RB NNP .

Bill Gates founded Microsoft.
NNP NNP VBD NNP .

Figner was a Russian revolutionary and narodnik born June 25, 1852 in Kazan, Russia.
NNP VBD DT JJ NN CC NN VBN NNP CD , CD IN NNP , NNP .

Dovlatov’s birthplace was Ufa, Republic of Bashkiria, USSR.
NNP POS NN VBD , NNP IN NNP , NNP .

Extractor

<Aksakov, was born in what is now, Bahk.>
<Garshin, was born in, Ekaterinoslav>
<Figner, was, a Russian ...>
<Dovlatov, 's birthplace was, Ufa>
## Types of Information Extraction

<table>
<thead>
<tr>
<th>Input</th>
<th>Relations</th>
<th>Traditional, Supervised IE</th>
<th>TextRunner &amp; WOE OpenIE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Corpus + Manual Labels</td>
<td>Corpus + Wikipedia/PennTB + Domain-Independent Methods</td>
</tr>
<tr>
<td></td>
<td>Specified in Advance</td>
<td></td>
<td>Uninterpreted Text Strings</td>
</tr>
</tbody>
</table>

- **<Aksakov, was born in what is now, Bahk.>**
- **<Garshin, was born in, Ekaterinoslav>**
- **<Figner, was, a Russian ...>**
- **<Dovlatov, ‘s birthplace was, Ufa>**
Jerome Allen “Jerry” Seinfeld is an American stand-up comedian, actor and writer, best known for playing a semi-fictional version of himself in the situation comedy Seinfeld (1989-1998), which he co-created and co-wrote with Larry David, and, in the show’s final two seasons, co-executive-produced.

Seinfeld was born in Brooklyn, New York. His father, Kalman Seinfeld, was of Galician Jewish background and owned a sign-making company; his mother, Betty, is of Syrian Jewish descent.

Name  Nationality
Bono Irish
Seinfeld American
Zhang Ziyi Chinese
### Types of Information Extraction

<table>
<thead>
<tr>
<th>Input</th>
<th>Traditional, Supervised IE</th>
<th>TextRunner &amp; WOE OpenIE</th>
<th>Kylin &amp; Luchs Weak Supervision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Corpus + Manual Labels</td>
<td>Corpus + Wikipedia/PennTB + Domain-Independent Methods</td>
<td>Corpus + Wikipedia/PennTB + Domain-Independent Methods</td>
</tr>
<tr>
<td>Relations</td>
<td>Specified in Advance</td>
<td>Uninterpreted Text Strings</td>
<td>Learned</td>
</tr>
</tbody>
</table>
Ayn Rand
From Wikipedia, the free encyclopedia
(Redirected from Ayn rand)

Ayn Rand (pronounced /əˈrnɛnd/; born Alisa Zinov'yevna Rosenbaum; February 2, 1905 – March 6, 1982), was a Russian-American novelist, philosopher, playwright, and screenwriter. She is known for her two best-selling novels and for developing a philosophical system she called Objectivism. Born and educated in Russia, Rand emigrated to the United States in 1926. She worked as a screenwriter in Hollywood and had a play produced on Broadway in 1935–1936. She first achieved fame in 1943 with her novel The Fountainhead, which in 1957 was followed by her best-known work, the philosophical novel Atlas Shrugged.

Rand's political views, reflected in both her fiction and her theoretical work, emphasize individual rights (including property rights) and laissez-faire capitalism, enforced by a constitutionally limited government. She was a fierce opponent of all forms of collectivism and statism,[4] including fascism, communism, socialism, and the welfare state,[5] and promoted ethical egoism while rejecting the ethic of altruism.[6] She considered reason to be the only means of acquiring knowledge and the most important aspect of her philosophy,[7] stating, “I am primarily an advocate of capitalism, but of egoism, and I am not primarily an advocate of egoism, but of reason. If one recognizes the supremacy of reason and applies it consistently, all the rest follows.”[8]

Contents [show]

Life and work

Early life
Rand was born Alisa Zinov'yevna Rosenbaum (Russian: Алиса Зиновьевна Розенбаум) on February 2, 1905, to a bourgeois family living in Saint Petersburg. She was the eldest of the three daughters (Alisa, Natasha, and Nora) of Zinovy Zakharovich Rosenbaum and Anna Borisovna Rosenbaum, largely non-observant Jews. Her father was educated as a chemist and became a successful pharmacist, eventually owning his own pharmacy and the building in which it was located.[9] His success allowed the family to employ a cook, maid, nurse, and governess.[10] Growing up, she was praised by adults for her intelligence, but her intensity and social awkwardness meant she rarely had friends her own age. On one occasion, when a school assignment called for her to write about the joys of childhood, she instead wrote what she later recalled as “a scathing denouncement of childhood as inferior to the intellectual capacity of adults”[11]

Rand was twelve at the time of the Russian Revolution of 1917. Opposed to the Tsar, Rand's sympathies were with Alexander Kerensky. Rand's family life was disrupted by the rise of the Bolshevik party under Vladimir Lenin. Her father's pharmacy was confiscated by the Bolsheviks, and the family fled to the Crimea, which was initially under the control of the White Army during the Russian Civil War. She later recalled that while in high school she determined that she was an atheist and that she valued reason and intellect. After graduating from high school in the Crimea she briefly held a job teaching Red Army soldiers to read. She found she enjoyed that work very much, the illiterate soldiers being eager to learn and respectful of her. At sixteen, Rand returned with her family to Saint Petersburg.[12][13]

Following the Russian Revolution, universities were opened to women, including Jews, allowing Rand to be in the first group of women to enroll at Petrograd State University,[14] where she studied in the department of social pedagogy, majoring in history.[15] At the university she was introduced to the writings of Aristotle and Plato, who would form two of the greatest influences and counter-influences respectively on her thought.[16] A third figure whose philosophical works she studied heavily was Friedrich Nietzsche.[17] Her formal study of philosophy amounted to only a few courses, and outside of these three philosophers, her study of key figures in philosophy was mainly self-taught.

Influences [show]

Influenced [show]

Signature

Ayn Rand

Born
Alisa Zinov'yevna Rosenbaum
February 2, 1905
Saint Petersburg, Russian Empire

Died
March 6, 1982 (aged 77)
New York City, United States

Occupation
Philosopher, writer

 Alma mater
University of Petrograd

Notable works
Atlas Shrugged

Holding on a reference to the database.
Wikipedia Infoboxes

- Thousands of relations encoded in infoboxes
- Infoboxes are interesting target:
  - By-product of thousands of contributors
  - Broad in coverage and growing quickly
  - Schema noisy and sparse, extraction is challenging
Jerome Allen “Jerry” Seinfeld is an American stand-up comedian, actor and writer, best known for playing a semi-fictional version of himself in the situation comedy Seinfeld (1989-1998), which he co-created and co-wrote with Larry David, and, in the show’s final two seasons, co-executive-produced.

Seinfeld was born in Brooklyn, New York. His father, Kalman Seinfeld, was of Galician Jewish background and owned a sign-making company; his mother, Betty, is of Syrian Jewish descent.
Existing work on Kylin

- Kylin performed well on popular classes ...
  *Precision: mid 70% ~ high 90%*
  *Recall: low 50% ~ mid 90%*
- ... but floundered on sparse classes.
  *(too little training data)*
- Is this a big problem?
- 82% have less than 100 instances
  40% have less than 10 instances
Problem: Sparse Attributes

Training Data:
A. was born in **London**.
O. was born in 1948.
C. was born in **Auckland**.

Test Data:
B. was born in **Auckland**.
G. was born in **Stockholm**.
T.’s birthplace was **Tokyo**.

If extractor had a list of cities, it could do better!
Need many, many Lists
New System

- **Luchs** – autonomous system – learns to extract 5025 relations
- **Luchs** introduces *dynamic lexicon features*
  - allow learning from sparse data
  - improves scalability
- **Luchs** achieves F1 > 80% on 1000+ relations
Outline

• Motivation
• **System Overview**
• Extraction with Dynamic Lexicons
• Experiments
Overview of LUCHS

Learning

Matcher

Training Data
Overview of LUCHS

Learning

Matcher

Training Data

Extractor Learner

Extraction

Attribute Extractor

thousands

Tuples
Infoboxes have Schemas
Overview of LUCHS

Learning

Matcher

Extractor Learner

Extractor

Extraction

Article Classifier

Attribute Extractor

Tuples
Overview of LUCHS

Learning

Matcher

Training Data

Classifier Learner

Extracted Data

Extractor Learner

Article Classifier

Classified Articles

Attribute Extractor

Tuples
Learning Extractors

• **Classifier**: multi-class classifier using features: words in title, words in first sentence, ...

• **Extractor**: linear-chain CRF predicting label for each word, using features: words, state transitions, capitalization, word contextualization, digits, dependencies, first sentence, lexicons, Gaussians

• Trained using Voted Perceptron algorithm

[Collins 2002, Freund and Schapire 1999]
Overview of LUCHS

Learning

Matcher → Harvester → WWW

Training Data → Classifier Learner

Training Data → CRF Learner

Filtered Lists → Lexicon Learner

Lexicons

Extraction

Article Classifier → Classified Articles

Attribute Extractor → Tuples
Outline

• Motivation
• System Overview
• Extraction with Dynamic Lexicons
• Experiments
Harvesting Lists from the Web

• Must extract and index lists prior to learning

• Lists extremely noisy: navigation bars, tag sets, spam links, long text; filtering steps necessary

• 49M lists containing 56M unique phrases
Semi-Supervised Learning of Lexicons

Generate lexicons specific to relation in 3 steps:

1. Extract seed phrases from training set

2. Expand seed phrases into a set of lexicons

3. Add lexicons as features to CRF
From Seeds to Lexicons

Similarity between lists using vector-space model:

Intuition: lists are similar if they have many overlapping phrases, the phrases are not too common, and lists are not too long.
From Seeds to Lexicons

Produce lexicons of different Pr/Re compromises:

Sort by similarity to seeds

Union of phrases on top lists
Semi-Supervised Learning of Lexicons

Generate lexicons specific to relation in 3 steps:

1. Extract seed phrases from training set
2. Expand seed phrases into a set of lexicons
3. Add lexicons as features to CRF
Preventing Lexicon Overfitting

• Lexicons created from seeds in training set
• CRF may overfit if trained on *same examples* that generated the lexicon features
With ... Cross-Training

• Lexicons created from seeds in training set
• CRF may overfit if trained on *same examples* that generated the lexicon features
• Split training set into *k partitions*
• Different partitions for lexicon creation, learning

---

Seinfeld was born in **Brooklyn** New York...
Born in **Omaha**, Tony later developed ...
His birthplace was **Boston** ...

His hometown **Denver** is well known for ...
where he was born is ...
Simon, born and grown up in **Seattle** ...

He was born in **Spokane**
Portland is his hometown.
Tony was born in  **Austin**
Outline

• Motivation
• System Overview
• Extraction with Dynamic Lexicons
• Experiments
Experiments

- English Wikipedia 10/2008 dump
- Classes with at least 10 instances: 1,583, comprising 981,387 articles
  5025 attributes
- Consider first 10 sentences of each article
- Evaluate extraction on a token-level
Overall Extraction Performance

Tested pipeline of classification and extraction:
• Compared to manually created gold labels
  – on 100 articles not used for training

Observations:
• Many remaining errors from “ontology” sloppiness
• low recall for heuristic matches

\[
\begin{array}{|c|c|}
\hline
\text{F1} & 0.61 \\
\text{P} & 0.55 \\
\text{R} & 0.68 \\
\hline
\end{array}
\]
Article Classification

• Take all 981,387 articles which have infoboxes 4/5 for training, 1/5 for testing,
  Use existing infobox as gold standard

• Accuracy: 92%

• Again, many errors due to “ontology” sloppiness:
  e.g. Infobox Minor Planet vs. Infobox Planet
Attribute Extraction

• For each of 100 attributes, sample 100 articles for training and 100 articles for testing
• Use heuristic matches as gold labels
• Baseline extractor: iteratively add feature with largest improvement (except lexicon & Gaussian)
Impact of Lexicons

Lexicons substantially improve F1

Cross-training essential

<table>
<thead>
<tr>
<th>Text attributes</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>.491</td>
</tr>
<tr>
<td>Baseline + lexicons w/o cross-training</td>
<td>.367</td>
</tr>
<tr>
<td>Baseline + lexicons w/ cross-training</td>
<td>.545</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Numeric attributes</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>.586</td>
</tr>
<tr>
<td>Baseline + Gaussians w/o cross-training</td>
<td>.623</td>
</tr>
<tr>
<td>Baseline + Gaussians w/ cross-training</td>
<td>.627</td>
</tr>
</tbody>
</table>
Impact on Sparse Attributes

<table>
<thead>
<tr>
<th># train. articles</th>
<th>ΔF1</th>
<th>ΔPrecision</th>
<th>ΔRecall</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Text attributes</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>+16%</td>
<td>+10%</td>
<td>+20%</td>
</tr>
<tr>
<td>25</td>
<td>+13%</td>
<td>+7%</td>
<td>+20%</td>
</tr>
<tr>
<td>100</td>
<td>+11%</td>
<td>+5%</td>
<td>+17%</td>
</tr>
<tr>
<td><strong>Numeric attributes</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>+10%</td>
<td>+4%</td>
<td>+13%</td>
</tr>
<tr>
<td>25</td>
<td>+8%</td>
<td>+4%</td>
<td>+10%</td>
</tr>
<tr>
<td>100</td>
<td>+7%</td>
<td>+5%</td>
<td>+8%</td>
</tr>
</tbody>
</table>

- Lexicons very effective for sparse attributes
- Gains mostly in recall
Scaling to all of Wikipedia

• Extract all 5025 attributes (matches as gold)
• 1138 attributes reach F1 score of .80 or higher

• Average F1 of .56 for text and .60 for numeric attr.
• Weighted by #instances, .64 and .78 respectively
Towards an Attribute Ontology

• True promise of relation-specific extraction if ontology ties system together
• Identify duplicates in infobox ontology
• Goal: Jointly learn ontology and extractors
Conclusions

• Weakly-supervised learning of relation-specific extractors *does scale*

• Introduced *dynamic lexicon features*, which enable hyper-lexicalized extractors

This material is based on work supported by a WRF / TJ Cable Professorship, a gift from Google and by Air Force Research Laboratory (AFRL) under prime contract # FA8750-09-C-0181. Any opinions, findings, and conclusion or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of the Air Force Research Laboratory (AFRL).
Related Work

– YAGO [Suchanek et al WWW 2007]
– Bayesian Knowledge Corroboration [Kasneci et al MSR 2010]
– PORE [Wang et al 2007]
– TextRunner [Banko et al IJCAI 2007]
– Distant Supervision [Mintz et al ACL 2009]
– Kylin [Wu et al CIKM 2007, Wu et al KDD 2008]