Learning to rank adaptively for scalable information extraction

**Pablo Barrio**, Columbia University
Gonçalo Simões, INESC-ID and IST, University of Lisbon
Helena Galhardas, INESC-ID and IST, University of Lisbon
Luis Gravano, Columbia University
Information Extraction (IE)

- Natural-language text **embeds** “structured” data
- Information extraction systems **extract** this data

“... A **tornado** swept the coast of **Florida** on Wednesday...”

Extracted tuple for **Natural Disaster-Location** relation

**Much richer querying and analysis possible**
IE is Challenging and Time Consuming

- Operates over large sets of features
  - Bag of words, N-grams, grammar productions, dependency paths
  - "... A tornado swept the coast of Florida on Wednesday..."
  - tornado swept swept the coast of Florida
  - May grow as large as number of unique words and sequences of N words

- Requires complex text analysis
  - Dependency parsing, entity recognition, syntactic parsing, shallow parsing, part-of-speech tagging, semantic role labeling

May take several seconds per document (e.g., with subsequence kernel extractor for Natural Disaster-Location)
Problematic over large document collections
Reducing Processing Time: Opportunities

Documents are “useful” if they produce output for a given IE task

- **Small, topic-specific** fraction of collection
  
  Only **2% of documents** in a New York Times archive, mostly **environment-related**, are useful for Natural Disaster-Location with a state-of-the-art IE system

- Useful documents share **distinctive words and phrases**
  
  “Earthquake,” “storm,” “Richter,” “volcano eruption” for Natural Disaster-Location

- Information extraction system “**labels**” documents as useful or not **for free**

  Should focus extraction over these documents and ignore rest

  Can learn to differentiate between useful documents for an IE task and rest

  IE process generates ever-expanding training set for learning to identify useful documents
Existing Approaches: QXtract and FactCrawl

QXtract and FactCrawl learn from small document sample and exhibit far-from-perfect recall.

FactCrawl ranks documents using learned queries and does not adapt to new processed documents.

[Eugene Agichtein and Luis Gravano, "Querying text databases for efficient information extraction." ICDE ’03]
[Christoph Boden et al., "FactCrawl: A fact retrieval framework for full-text indices." WebDB ’11]
Our Approach: Key Aspects

- Document ranking needs to be **robust and efficient**
  - Learning to rank approach for document ranking

- Results of extraction process form **ever-expanding training set**
  - Adaptive approach to update document ranking continuously
Ranking Documents Adaptively for IE

Learns that “tornado,” “earthquake,” or “aftermath” are markers of useful documents.

Learning

Document processing and update detection

Useful documents but on volcanoes, not yet observed prominently in IE process.

New information can potentially help improve ranking, so Update!

Online relearning

Learns that “volcano” and “eruption” are now markers of useful documents.
Ranking Documents Adaptively for IE: Our Alternatives

• Efficient learning-to-rank techniques for information extraction: BAgg-IE, RSVM-IE

• Update detection techniques for document ranking adaptation: Top-K, Mod-C
Efficient Learning to Rank for IE: BAgg-IE

- Based on **bootstrapping aggregation**

**Learning Algorithm**

- **Bootstrapping**: Randomly w/o replacement
- **Training**: Binary SVM classifiers

**Ranking Model**

- **Binary Classifier**
- **Scoring**: Normalized score
- **Ranking and processing**
- **Aggregation**: Sum of scores

**Words and phrases that make a document useful**

- tornado, swept
- aftermath
- earthquake, tornado

**More stable classification**

**Relevant Features**

- tornado, swept
- aftermath
- earthquake, tornado

**All models are trained using online learning and in-training feature selection**
Efficient Learning to Rank for IE: RSVM-IE

- Based on **RankSVM**
  - Learns SVM classifier on pairwise difference of documents

<table>
<thead>
<tr>
<th>Learning model</th>
<th>Training instance</th>
</tr>
</thead>
<tbody>
<tr>
<td>RankSVM</td>
<td>$d_i - d_n$</td>
</tr>
<tr>
<td>SVM</td>
<td>$d_i$</td>
</tr>
</tbody>
</table>

Training Label is 1 iff $d_i$ is “better” than $d_n$

Model is trained using online learning and in-training feature selection
Ranking Documents Adaptively for IE: Our Alternatives

- Efficient learning-to-rank techniques for information extraction: BAgg-IE, RSVM-IE

- Update detection techniques for document ranking adaptation: Top-K, Mod-C
Update Detection for Document Ranking Adaptation: Top-\(K\)

- Uses only **most important** (top-\(K\)) features

**Done during document ranking**

Find top-\(K\) features

- Document Sample
- Binary SVM Classifier

**Done during document processing**

Detect feature changes

- Document Sample
- Processed Documents
- Binary SVM Classifier

Weights indicate importance

- \(\text{storm, 3.2 richter, 2.8} \)
- \(\ldots \text{people, 0.1} \)

**Generalized Spearman’s Footrule**

- \(\text{richter, 3 eruption, 2.9} \)
- \(\ldots \text{people, 0.06} \)

\( \geq \tau \) ? Update Ranking
Update Detection for Document Ranking Adaptation: Mod-C

- Uses all features

Obtain features

Detect feature changes

Features recovered from model

Cosine Similarity $\cos(\theta) > \tau$? Update Ranking

Obtained “for free” during document ranking

Done during document processing
Experimental Settings

- Information extraction systems

**Simple extraction systems:**
- HMMs, text patterns

**Complex extraction systems:**
- CRFs, SVM kernels

---

**Person-Organization**

<table>
<thead>
<tr>
<th>Person</th>
<th>Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Larry Page</td>
<td>Google</td>
</tr>
<tr>
<td>Sergey Brin</td>
<td>Google</td>
</tr>
</tbody>
</table>

**Disease-Outbreaks**

- The Haiti cholera outbreak between 2010 and 2013 was the worst epidemic of cholera in recent history.

<table>
<thead>
<tr>
<th>Disease</th>
<th>Time Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>cholera</td>
<td>between 2010 and 2013</td>
</tr>
</tbody>
</table>

**Person-Career**

- "This is not a victimless crime," said Jim Kendall, president of the Washington Association of Internet Service Providers.

<table>
<thead>
<tr>
<th>Person</th>
<th>Career</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jim Kendall</td>
<td>President</td>
</tr>
</tbody>
</table>

**Man Made Disaster-Location**

<table>
<thead>
<tr>
<th>Disaster</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>fire</td>
<td>Booneville</td>
</tr>
</tbody>
</table>

A fire destroyed a Cargill Meat Solutions beef processing plant in Booneville.

**Other relations:**
- Person-Charge, Election-Winner, Natural Disaster-Location
Does Learning Ranking Models Help?

- **Learning ranking models** leads to better document ranking
- **RSVM-IE** performs best at *early stages*
- **BAgg-IE** obtains high gains *later on*
- Objective function of **learning model** shapes document ranking

Additional experiments in paper: analogous conclusions over all relations
Does Update Detection Help?

- **Feat-S** unable to evaluate over new features, crucial during adaptation
- **Top-K** and **Mod-C** improve the efficiency of the extraction process
- **Mod-C** leads to best execution using more efficient approach, with fewer models

Additional experiments in paper: analogous conclusions over all relations
Putting Learning to Rank and Update Detection Together: Recall Analysis

- **Our techniques** bring **significant improvement** for sparse relations
- **RSVM-IE** performs best, as it prioritizes useful documents better, favoring adaptation

Additional experiments in paper: analogous conclusions over all relations
Putting Learning to Rank and Update Detection Together: Extraction Time

- Cost of adapting in A-FactCrawl hurts efficiency of extraction process
- **Our techniques improve** efficiency of process even for inexpensive IE systems

Additional experiments in paper for **our techniques**:  
- Analogous conclusions also for expensive IE systems and sparse relations  
- **Scale linearly** in the size of the collection
Document Ranking for Scalable Information Extraction: Summing Up

- Running IE system over large text collections is computationally expensive

- Proposed lightweight, adaptive approach and learning-based alternatives

  - Online learning algorithms with in-training feature selection: RSVM-IE, BAgg-IE

  - Update detection based on feature changes: Mod-C, Top-K

- RSVM-IE + Mod-C performs best: Useful documents are better prioritized, enabling richer, more efficient ranking adaptation
Future Work: Ranking at Different Granularities

- **Few collections** on the Web are relevant to an IE task
  
  Prioritize them based on number of useful documents

- **Few sentences** in a text document output tuples for an IE task
  
  Prioritize them based on usefulness and diversity
Future Work: Distributing the Execution of IE Systems

- Identify **optimal distributed execution strategy**
  
  E.g., by determining document placement in distributed file system
But Before We Leave…

Try **REEL**, our toolkit to easily develop and evaluate IE systems

Thanks!
## Information Extraction: Time Analysis

<table>
<thead>
<tr>
<th>Task</th>
<th>Time per sentence (ms)</th>
<th>Toolkit or Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentence splitting</td>
<td>0.1</td>
<td>PTB</td>
</tr>
<tr>
<td>Tokenization</td>
<td>0.1</td>
<td>PTB</td>
</tr>
<tr>
<td>Part-of-speech tagging</td>
<td>7.4</td>
<td>ClearNLP</td>
</tr>
<tr>
<td>Shallow parsing</td>
<td>42</td>
<td>Search</td>
</tr>
<tr>
<td>Dependency parsing</td>
<td>25.6</td>
<td>ClearNLP</td>
</tr>
<tr>
<td>Semantic role labeling</td>
<td>8.4</td>
<td>ClearNLP</td>
</tr>
<tr>
<td>Named Entity recognition (per entity)</td>
<td>1.1</td>
<td>SENNA</td>
</tr>
<tr>
<td>Relation extraction</td>
<td>766 67</td>
<td>Tree Kernel OLLIE</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>850.7 151.7</strong></td>
<td></td>
</tr>
</tbody>
</table>
Experimental Settings: Data and Relations

- **Dataset:** *The New York Times* 1.8 million articles from 1987-2007
- **Information Extraction Systems**
  - **Simple extraction systems:** HMMs, Text patterns
  - **Complex extraction systems:** CRFs, SVM Kernels

### Dense relations

**Person-Organization**
- **Google** co-founders *Larry Page* and *Sergey Brin* recently sat down with billionaire venture capitalist *Vinod Khosla* for a lengthy interview.

**Person-Career**
- "This is not a victimless crime," said *Jim Kendall*, president of the Washington Association of Internet Service Providers.

### Sparse relations

**Disease-Outbreaks**
- The Haiti cholera outbreak between *2010* and *2013* was the worst epidemic of *cholera* in recent history.

**Man Made Disaster-Location**
- A *fire* destroyed a Cargill Meat Solutions beef processing plant in *Booneville*.

**Person-Charge**
- *Ibrahim Muktar Said* was charged Sunday night in connection with the failed Hackney bus *bombing*.

**Election-Winner**
- *Boris Johnson* defeated *Ken Livingstone* in the *London mayoral election*.

**Natural Disaster-Location**
- A *tornado* swept the coast of *Florida* on Wednesday.
Experimental Settings: Extractors

• Person-Organization Affiliation:
  • Entities: HMM and text patterns
  • Relation: SVM classifier

• Disease-Outbreak:
  • Entities: Dictionaries and manually crafted regular expressions
  • Relation: Distance between entities

• Others:
  • Entities: Stanford NLP (Person and Location), MEMM (Natural Disasters), and CRF (others)
  • Relation: Subsequences Kernel [Bunescu and Mooney, NIPS '05]
Experimental Settings: Details

• Document Sampling Strategies:
  
  • Simple Random Sampling (SRS): Documents are collected randomly from fully-accessible collection
  
  • Cyclic Querying Sampling (CQS): Queries learned from external collection and issued in a round-robin fashion
  
• Update Detection:

  • Feature Shifting (Feat-S): Gaussian kernel for one-class classification
    • Triggers an update for high geometrical difference
      [A. Glazer, "Feature Shift Detection." ICPR ’12]

  • Fixed Window (Wind-F): Triggers after processing N documents
Ranking Models vs. FactCrawl

- Using **full document contents** leads to better document ranking
- RSVM-IE performs best at **early stages**
- BAgg-IE obtains high gains **later on**
- **Objective function shapes the document ranking**
Impact of Document Sampling

- CQS improves recall at early stages
- CQS obtains higher average precision and AUC
- Targeted sampling improves the efficiency of the extraction process
Update Detection: Time and Distribution of Updates

- Wind-F is the **most efficient** but ignores document contents
- Feat-S performs fewer updates but is affected by kernel cost
- Top-K performs the **fewest** updates, relatively efficiently
- Mod-C exhibits best **number of updates-time balance**

Update detection baselines:
- **Wind-F** = Updates after processing 20,000 documents (2% of collection)
- **Feat-S** = Update method based on Gaussian kernel [Glazer, ICPR ’12]

**Average CPU time per document (ms):**
- Wind-F: 0.01
- Feat-S: 5.72
- Top-K: 1.89
- Mod-C: 0.32
Scalability Analysis: Running Time

- Our approach:
  - **Scales linearly** to collection size
  - **Improves** with the more information we find in larger collections
  - Is a substantial step towards scalable information extraction