Introduction to Information Extraction – 2

Wenjie Li (@ HK PolyU)
Tuesday, August 2, 2011, @HIT
Relation Extraction

• Relation Extraction
  – Definition
    • Predict the relationship between a given entity pair \((C, E_1, E_2) \rightarrow r (r \in R)\).
  – Approaches
    • Single-Label Multi-Class Classification Problem
    • Supervised Models: ME, SVM, ...
  – Feature-based vs. Kernel-based Approaches
    • Flat Features vs. Graph Structures
Relation Extraction

• Relation Extraction
  – Domains
    • News Articles: e.g. Location_of, Position_of, Product_of, ...
    • Scientific Publications (Bio-medical): e.g., Gene-Disease Relations, Protein–Protein Interaction, ...
      – BioCreAtIvE (Critical Assessment of Information Extraction systems in Biology)
    • Blogs, Emails, Wikipedia and General Web
Relation Extraction – Approaches

• Feature-based Approaches
  – Feature Sets
    • Entity Types/Sub-Types
    • Surface Words: Words around and in-between the Two Entities.
      \[
      \langle \text{Company} \rangle \text{ Kosmix} \langle /\text{Company} \rangle \text{ is located in the}\n      \langle \text{Location} \rangle \text{ Bay area} \langle /\text{Location} \rangle.\n      \]
    • Part-of-Speech (POS) Tags: Verbs are key to defining the relationship between entities that are typically nouns or noun phrases.
Relation Extraction – Approaches

There are several approaches to relation extraction, including rule-based methods, machine learning models, and deep learning techniques. Rule-based methods rely on predefined rules to extract relationships, while machine learning models learn from labeled data to identify relationships. Deep learning techniques, such as convolutional neural networks and recurrent neural networks, have been shown to be effective in extracting relationships from text.

An example of a unigram feature is:

[String = “host”, flag = “none”]
[Part of speech = Verb, flag = “none”].

Example bigram features are:

[Strings = “(host, ICML)”, flags = “(none,2)”, type = “sequence”]
[Part of speech = (Verb,Noun) flag = “(none,2)”, type = “sequence”]
[(String = “host”, Part of speech = Noun), flag = “(none,2)”, type = “sequence”].

Example trigram features are:

[Strings = “(will, host, ICML)”, flags = “(none,none,2)”, type = “sequence”]
[Part of speech = (Modifier, Verb,Noun) flag = “(none,none,2)”, type = “sequence”].
Relation Extraction – Approaches

• Kernel-based Approaches
  – Design special kernels to capture the similarity between structures such as dependency trees or parse tree.

\[
\text{The University of Helsinki hosts ICML this year.}
\]

\[
\text{Haifa, located 53 miles from Tel Aviv will host ICML in 2010.}
\]
Relation Extraction – Approaches

• Kernel-based Approaches
  – Examples of Syntactic Parse Tree and Dependency Tree

```
(ROOT
  (S
    (NP (NNP Haifa))
    (VP (VBN located)
      (PP
        (NP (CD 53) (NNS miles))
        (IN from)
        (NP (NNP Tel) (NNP Aviv)))))
    (VP (MD will)
      (VP (VB host)
        (NP
          (NP (NNP ICML))
          (PP (IN in)
            (NP (CD 2010)))))))
```
Relation Extraction – Approaches

• Kernel-based Approaches

  – Kernel Functions

  • Encode the similarity between the two graphs as a kernel function.

  \[ X = (x, E_1, E_2) \]

  \[ \hat{r} = \arg\max_{r \in \mathcal{Y}} \sum_{i=1}^{N} \alpha_{ir} K(X_i, X) \]

• Example 1: A Shortest Path based Kernel on Dependency Trees

\[ K(P, P') = \begin{cases} 0 & \text{if } P, P' \text{ have different lengths} \\ \lambda \prod_{k=1}^{\lfloor P \rfloor} \text{CommonProperties}(P_k, P'_k) & \text{otherwise,} \end{cases} \]
Relation Extraction – Approaches

• Kernel-based Approaches
  – Kernel Functions
    • Example 2: A Convolution Tree Kernel on Parse Trees

\[ K_{CTK}(T_1, T_2) = \sum_{n_1 \in N_1, n_2 \in N_2} \Delta(n_1, n_2) \]

where \( N_1 \) and \( N_2 \) are the sets of nodes in trees \( T_1 \) and \( T_2 \), respectively.

  – \( \Delta(n_1, n_2) \) can be computed recursively.
Relation Extraction – Approaches

- Kernel-based Approaches
  - Kernel Functions
    - Example 2: A Convolution Tree Kernel on Parse Trees

1) If the context-free productions (i.e., context-free grammar rules) at \( n_1 \) and \( n_2 \) do not match, then \( \Delta(n_1, n_2) = 0 \); otherwise go to 2).

2) If both \( n_1 \) and \( n_2 \) are POS tags, then \( \Delta(n_1, n_2) = \lambda \); otherwise (i.e., if both \( n_1 \) and \( n_2 \) are constituent tags, such as NP and S) go to 3).

3) Calculate \( \Delta(n_1, n_2) \) recursively as:

\[
\Delta(n_1, n_2) = \lambda \prod_{k=1}^{\#ch(n_1)} (1 + \Delta(ch(n_1, k), ch(n_2, k)))
\]

where \#ch(n) is the number of children of node n, \( ch(n, k) \) is the \( k \)-th child of node n and \( \lambda (0 < \lambda < 1) \)
Relation Extraction – Approaches

• Feature-based vs. Kernel-based Approaches
  – Tree Structure -> Structure Features

\[
\text{[(Entity label = Location, Part of speech = Verb), flag = "(1, none)", type = "dependency"]}.
\]

This feature fires on node pair “(Haifa, host)” connected as “host ← Haifa” in the dependency graph.

\[
\text{[(POS = (noun,verb,noun), flag = "(1,none,2)", type = "dependency"]}.
\]

This feature fires on the nodes “(Haifa, host, ICML)” because of the edge pattern: “Haifa → host ← ICML.”

Haifa located 53 miles from Tel Aviv will host ICML in 2010
Relation Extraction – Approaches

• Feature Space Exploration
  – A Systematic Exploration of the Feature Space for Relation Extraction (NAACL/HLT 2007)
    • http://sifaka.cs.uiuc.edu/czhai/pub/hlt07-rel.pdf
  – Relation Instance Representations
    • Sequence

0 indicates that a node does not cover any argument, 1 or 2 indicates that the node covers arg1 or arg2, respectively, and 3 indicates that the covers both arguments.
Relation Extraction – Approaches

• Feature Space Exploration
  – Relation Instance Representations
    • Syntactic Parse Tree
Relation Extraction – Approaches

• Feature Space Exploration
  – Relation Instance Representations
  • Dependency Parse Tree
Relation Extraction – Approaches

• Feature Space Exploration
  – Feature Spaces
    • Entity Attribute Features: Entity Types
    • Bag-of-Word Features: Word N-Grams
    • Grammar Production Features
    • Dependency Relation and Dependence Path Features
Relation Extraction – Approaches

• Feature Space Exploration
  – Feature Sub-Spaces
    • Sequence
    • Syntactic Parse Tree
    • Dependence Parse Tree
  – In each sub-space, explores
    • Uni-Gram Features: Single Node with Single Attribute
    • Bi-Gram Features: Single Edge Connecting Two Nodes
    • Tri-gram Features: Two Connected Edges and Three Nodes
## Relation Extraction – Approaches

- **Feature Space Exploration**  
  – Experimental Comparison

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>Uni</th>
<th>+Bi</th>
<th>+Tri</th>
<th>+Prod</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ME</strong></td>
<td><strong>Seq</strong></td>
<td>0.647</td>
<td>0.662</td>
<td>0.717</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>0.645</td>
<td>0.698</td>
<td>0.726</td>
<td>0.702</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.648</td>
<td>0.695</td>
<td>0.707</td>
<td>0.696</td>
<td></td>
</tr>
<tr>
<td><strong>Syn</strong></td>
<td>R</td>
<td>0.651</td>
<td>0.697</td>
<td>0.726</td>
<td>0.702</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>0.648</td>
<td>0.697</td>
<td>0.707</td>
<td>0.696</td>
</tr>
<tr>
<td><strong>Dep</strong></td>
<td>R</td>
<td>0.647</td>
<td>0.673</td>
<td>0.718</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>0.647</td>
<td>0.673</td>
<td>0.718</td>
<td>N/A</td>
</tr>
<tr>
<td><strong>SVM</strong></td>
<td><strong>Seq</strong></td>
<td>0.583</td>
<td>0.666</td>
<td>0.684</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>0.586</td>
<td>0.650</td>
<td>0.684</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.585</td>
<td>0.658</td>
<td>0.684</td>
<td>N/A</td>
<td></td>
</tr>
<tr>
<td><strong>Syn</strong></td>
<td>R</td>
<td>0.598</td>
<td>0.645</td>
<td>0.679</td>
<td>0.674</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>0.611</td>
<td>0.663</td>
<td>0.681</td>
<td>0.672</td>
</tr>
<tr>
<td></td>
<td>0.604</td>
<td>0.654</td>
<td>0.680</td>
<td>0.673</td>
<td></td>
</tr>
<tr>
<td><strong>Dep</strong></td>
<td>R</td>
<td>0.583</td>
<td>0.644</td>
<td>0.682</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>F</td>
<td>0.583</td>
<td>0.644</td>
<td>0.682</td>
<td>N/A</td>
</tr>
</tbody>
</table>

- Precision (P), Recall (R), and F-score (F) values are shown.
Relation Extraction – Approaches

• Feature Space Exploration
  – Experimental Comparison

<table>
<thead>
<tr>
<th></th>
<th>Seq+Syn</th>
<th>Seq+Dep</th>
<th>Syn+Dep</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>ME</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>0.737</td>
<td>0.687</td>
<td>0.695</td>
<td>0.724</td>
</tr>
<tr>
<td>R</td>
<td>0.694</td>
<td>0.682</td>
<td>0.731</td>
<td>0.702</td>
</tr>
<tr>
<td>F</td>
<td>0.715</td>
<td>0.684</td>
<td>0.712</td>
<td>0.713</td>
</tr>
<tr>
<td>SVM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>0.689</td>
<td>0.669</td>
<td>0.687</td>
<td>0.691</td>
</tr>
<tr>
<td>R</td>
<td>0.686</td>
<td>0.653</td>
<td>0.682</td>
<td>0.686</td>
</tr>
<tr>
<td>F</td>
<td><strong>0.688</strong></td>
<td>0.661</td>
<td>0.684</td>
<td><strong>0.688</strong></td>
</tr>
</tbody>
</table>

• Conclusions
  – Using only basic unit features within each feature subspace is generally sufficient to achieve state-of-art performance, while over-inclusion of complex features might hurt the performance.
Relation Extraction – Approaches

• Feature Space Exploration
  – Experimental Comparison
  • Conclusions
    – When the three sub-spaces are combined, the performance can improve only slightly, which suggests that the sequence, syntactic and dependency relations have much overlap for the task of relation extraction.
Relation Extraction – Approaches

• Relation Extraction
  – Our Experience in ACE 2005 Chinese Relation Detection and Characterization (RDC) Task
     • A Novel Feature-based Approach to Chinese Entity Relation Extraction (ACL’2008)
     • [http://www.aclweb.org/anthology-new/P/P08/P08-2023.pdf](http://www.aclweb.org/anthology-new/P/P08/P08-2023.pdf)
     • Developing Position Structure based Framework for Chinese Entity Relation Extraction (ACM TALIP 2011)
     • Feature-based Approach
Relation Extraction – Our Experience

• Position Features
  – Observations
  • In many Part-Whole relation instances, one entity is often nested in the other entity.


Nested Structure with Relation Type/Subtype: PART-WHOLE/ Subsidiary
Relation Extraction – Our Experience

• Position Features
  – Observations
  • For many Physical/Located relations, the two entities are more likely to be adjacent (i.e., not nested and no entity in between)

[Thousands of Palestinians] rushed [the Israeli checkpoint].

Adjacent Structure with Relation Type/Subtype: PHYS/Located
Relation Extraction – Our Experience

- Position Features
  - Nine Position Structures

(a) Nested

(b) Nested-Nested

(c) Superposition

(d) Adjacent

(e) Nested-Adjacent

(f) Nested-Nested-Adjacent

(g) Separated

(h) Nested-Separated

(i) Nested-Nested-Separated
Relation Extraction – Our Experience

• Position Features
  – Imbalance Problem

<table>
<thead>
<tr>
<th>Structure types</th>
<th>#Positive class</th>
<th>#Negative class</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nested+</td>
<td>6332</td>
<td>4612</td>
<td>1 : 0.7283</td>
</tr>
<tr>
<td>Adjacent+</td>
<td>2028</td>
<td>27100</td>
<td>1 : 13.3629</td>
</tr>
<tr>
<td>Separated+</td>
<td>939</td>
<td>79989</td>
<td>1 : 85.1853</td>
</tr>
<tr>
<td>Overall</td>
<td>9299</td>
<td>111701</td>
<td>1 : 12.01</td>
</tr>
</tbody>
</table>

The ratios of positive to negative relation instances on 3-position structures

<table>
<thead>
<tr>
<th>Structure Types</th>
<th>#Positive Class</th>
<th>#Negative Class</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nested</td>
<td>6325</td>
<td>2347</td>
<td>1 : 0.37</td>
</tr>
<tr>
<td>Adjacent</td>
<td>1978</td>
<td>13501</td>
<td>1 : 6.82</td>
</tr>
<tr>
<td>Separated</td>
<td>928</td>
<td>39808</td>
<td>1 : 42.87</td>
</tr>
<tr>
<td>Superposition</td>
<td>6</td>
<td>407</td>
<td>1 : 67.84</td>
</tr>
<tr>
<td>Nested-Nested-Adjacent</td>
<td>50</td>
<td>3480</td>
<td>1 : 69.60</td>
</tr>
<tr>
<td>Nested-Nested-Separated</td>
<td>10</td>
<td>9142</td>
<td>1 : 914.20</td>
</tr>
<tr>
<td>Nested-Nested</td>
<td>1</td>
<td>1858</td>
<td>1 : 1858.00</td>
</tr>
<tr>
<td>Nested-Adjacent</td>
<td>0</td>
<td>10119</td>
<td>1 : INF</td>
</tr>
<tr>
<td>Nested-Separated</td>
<td>1</td>
<td>31039</td>
<td>1 : 31039.00</td>
</tr>
<tr>
<td>Overall</td>
<td>9299</td>
<td>111701</td>
<td>1 : 12.01</td>
</tr>
</tbody>
</table>

The ratios of positive to negative relation instances on 9-position structures
Relation Extraction – Our Experience

• Relation Classification
  – Feature Sets
    • Entity Type and Subtype Features
    • Uni-gram and Bi-gram Features (Entity Internal Character Sequence, In-Between Character Sequence, and External Context Character Sequence*)
    • Wordlist-based Features (Chinese Preposition List (e.g., 在, 朝, 往), orientation List (e.g., 东, 南, 西, 北), Auxiliary List (e.g., 的, 地, 得), and Conjunction List (e.g., 而且))
  – SVM Model
Relation Extraction – Our Experience

• ACE 2005 Dataset
  – 9,299 Relation Instances
  – 6 Relation Types and 18 Sub-Types

• 3-Position vs. 9-Position Features

<table>
<thead>
<tr>
<th>Types/Sub-Types</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-Position Baseline</td>
<td>73.06/71.54</td>
<td>34.84/31.27</td>
<td>47.18/43.52</td>
</tr>
<tr>
<td>9-Position Baseline</td>
<td>72.65/72.51</td>
<td>45.21/39.91</td>
<td>55.73/51.48</td>
</tr>
<tr>
<td>9-Position Partition</td>
<td>77.39/75.00</td>
<td>57.31/54.91</td>
<td>65.85/63.40</td>
</tr>
</tbody>
</table>
Relation Extraction – Our Experience

• ACE 2005 Dataset

<table>
<thead>
<tr>
<th>Relation Type</th>
<th>Relation Subtype</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>ART (Total No: 630)</td>
<td>User-Owner-Inventor-Manufacturer</td>
<td>630</td>
</tr>
<tr>
<td>GEN-AFF (Total No: 1937)</td>
<td>Citizen-Resident-Religion-Ethnicity</td>
<td>746</td>
</tr>
<tr>
<td></td>
<td>Org-Location</td>
<td>1191</td>
</tr>
<tr>
<td></td>
<td>Employment</td>
<td>1584</td>
</tr>
<tr>
<td></td>
<td>Founder</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>Ownership</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>Student-Alum</td>
<td>72</td>
</tr>
<tr>
<td>ORG-AFF (Total No: 2198)</td>
<td>Sports-Affiliation</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>Investor-Shareholder</td>
<td>85</td>
</tr>
<tr>
<td></td>
<td>Membership</td>
<td>346</td>
</tr>
<tr>
<td>PART-WHOLE (Total No: 2286)</td>
<td>Artifact</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Geographical</td>
<td>1289</td>
</tr>
<tr>
<td></td>
<td>Subsidiary</td>
<td>983</td>
</tr>
<tr>
<td>PER-SOC (Total No: 660)</td>
<td>Business</td>
<td>188</td>
</tr>
<tr>
<td></td>
<td>Family</td>
<td>384</td>
</tr>
<tr>
<td></td>
<td>Lasting-Personal</td>
<td>88</td>
</tr>
<tr>
<td>PHYS (Physical) (Total No: 1588)</td>
<td>Located</td>
<td>1358</td>
</tr>
<tr>
<td></td>
<td>Near</td>
<td>230</td>
</tr>
</tbody>
</table>
Relation Extraction – Our Experience

• All-at-Once Strategy vs. Cascade Strategy
  – Detect and Recognize in One Step
  – Detect and Recognize in Two Steps

• Relieve Class Imbalance Problem

<table>
<thead>
<tr>
<th>Types/Sub-Types</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>All-at-Once</td>
<td>77.39/75.00</td>
<td>57.31/54.91</td>
<td>65.68/63.40</td>
</tr>
<tr>
<td>Cascade</td>
<td>74.48/71.99</td>
<td>60.20/58.19</td>
<td><strong>66.58/64.36</strong></td>
</tr>
</tbody>
</table>
Relation Extraction – Our Experience

- Constraint(Knowledge)-based Correction
  - ACE Possible Relations

<table>
<thead>
<tr>
<th></th>
<th>PER</th>
<th>ORG</th>
<th>GPE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Per-Social.Family</td>
<td>Org-Aff.Employment, Ownership,</td>
<td>Physical.Near,</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>Org-Aff.Student-Alum,</td>
<td>Org-Aff.Employment</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Org-Aff.Sports-Affiliation</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ORG</td>
<td></td>
<td>Part-Whole.Subsidiary,</td>
<td>Part-Whole.Subsidiary,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPE</td>
<td></td>
<td>Org-Aff.Investor-Shareholder,</td>
<td>Physical.Near,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Org-Aff.Membership</td>
<td>Part-Whole.Geographical</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Relation Extraction – Our Experience

• Constraint(Knowledge)-based Correction
  – Wi/Wo Constraint(Knowledge)-based Correction

<table>
<thead>
<tr>
<th>Types/Sub-Types</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cascade</td>
<td>74.48/71.99</td>
<td>60.20/58.19</td>
<td>66.58/64.36</td>
</tr>
<tr>
<td>Cascade + Rectify + Adjust</td>
<td>76.58/77.48</td>
<td>61.90/60.19</td>
<td>68.46/67.75</td>
</tr>
</tbody>
</table>
Relation Extraction – Our Experience

• Consistency-based Correction
  – Guiding Top-Down: Type removes inconsistent subtype.
  – Strictly Bottom-Up: Subtype determines type.
  – Subtype Selection: Type selects subtype from most likely subtypes based on the probabilities of the classification results and decides if removes.
  – Type Selection: Subtype selects type from most likely types based on the probabilities of the classification results.
Relation Extraction – Our Experience

• Consistency-based Correction
  – Cascade + Rectify + Adjust

<table>
<thead>
<tr>
<th>Types/Sub-Types</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cascade + Rectify + Adjust</td>
<td>76.58/77.48</td>
<td>61.90/60.19</td>
<td>68.46/67.75</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Types/Sub-Types</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guiding Top-Down</td>
<td>77.48/77.91</td>
<td>61.88/58.83</td>
<td>68.81/67.04</td>
</tr>
<tr>
<td>Subtype Selection</td>
<td>79.47/77.52</td>
<td>61.76/59.23</td>
<td>69.50/67.15</td>
</tr>
<tr>
<td>Strictly Bottom-Up</td>
<td>80.38/77.44</td>
<td>62.45/60.16</td>
<td>70.29/67.71</td>
</tr>
<tr>
<td>Type Selection</td>
<td>80.81/78.06</td>
<td>62.31/60.04</td>
<td><strong>70.36/67.86</strong></td>
</tr>
</tbody>
</table>
Relation Extraction – Our Experience

• Co-reference and Linguistic Pattern based Inference

  – Co-reference based Inference

  • Motivation

    – Given $e_1=\{\text{He, Bill Gates}\}$, $e_2=\{\text{Microsoft}\}$, if $R=\{\text{Bill Gates, Microsoft}\}=\text{ORG-AFFILIATION}$, then $R=\{\text{He, Microsoft}\}=\text{ORG-AFFILIATION}$.

  • Strategy

    – Given two entities $e_1=\{em_{11}, em_{12}, ..., em_{1n}\}$ and $e_2=\{em_{21}, em_{22}, ..., em_{2m}\}$ ($e_i$ is an entity, $em_{ij}$ is a mention of $e_i$), it is true that $R(em_{11}, em_{21}) \Rightarrow R(em_{1l}, em_{2k})$
Relation Extraction – Our Experience

• Co-reference and Linguistic Pattern based Inference

  – Co-reference based Inference

    • Strategy

      – This nature allows us to infer more relations which may not be identified by classifiers.

      – Influence Order: Nested -> Adjacent -> Separated.

    • Sometimes, $R(em_{11}, em_{21}) \neq R(em_{12}, em_{22})$

    • Co-reference can also provides the second chance to check the consistency among relation mentions.
Relation Extraction – Our Experience

• Co-reference and Linguistic Pattern based Inference
  – Pattern based Inference
    • Motivation
      – Given “both $em_1$ and $em_2$ are located in $em_3$”, if $em_2$ and $em_3$ are classified as a positive relation instance, $em_1$ and $em_2$ will have the same relation type/subtype as that $em_2$ and $em_3$ hold.
    • Strategy
      – “And” and “Or” Patterns
Relation Extraction – Our Experience

- Co-reference and Linguistic Pattern based Inference

<table>
<thead>
<tr>
<th>Types/Sub-Types</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type Selection</td>
<td>80.81/78.06</td>
<td>62.31/60.04</td>
<td>70.36/67.86</td>
</tr>
<tr>
<td>+ Inference</td>
<td>80.71/77.75</td>
<td>62.48/60.20</td>
<td>70.43/67.86</td>
</tr>
</tbody>
</table>

- Doesn’t work as expected.
- Annotation doesn’t concern.
Relation Extraction – Our Experience

• Other Chinese Relation Extraction Work
  – Study of Kernel-Based Methods for Chinese Relation Extraction (AIRS 2008)
    • Convolution Tree Kernel
    • Shortest Path Dependency Kernel

<table>
<thead>
<tr>
<th>Types</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best Parse Tree Kernel</td>
<td>41.59</td>
<td>34.75</td>
<td>36.72</td>
</tr>
<tr>
<td>Best Dependency Kernel</td>
<td>17.89</td>
<td>18.62</td>
<td>15.20</td>
</tr>
<tr>
<td>Our Feature-based</td>
<td>80.71</td>
<td>62.48</td>
<td>70.43</td>
</tr>
</tbody>
</table>
Relation Extraction – Our Experience

• An Attempt to Apply Kernel-based Approach
  – Encode the Position Information as a Tree
    • Use parent/child nodes the nested relation between the two entities
    • Define the order among child nodes in terms of the appearance order of the corresponding entities in the sentence. In such a way, adjacent and separated relations are to the entity nodes under the same parent entity.
Relation Extraction – Our Experience

• An Attempt to Apply Kernel-based Approach
  – Encode the Position Information as a Tree
Relation Extraction – Our Experience

• An Attempt to Apply Kernel-based Approach
  – Entity Tree Kernel $K(T_1, T_2)$
    • Matching Function $m(t_i, t_j)$
      $$m(t_i, t_j) = \begin{cases} 1 & \text{if } t^m_i = t^m_j \\ 0 & \text{otherwise} \end{cases}$$
    • Similarity Function $s(t_i, t_j)$
      $$s(t_i, t_j) = \sum_{v_q \in t_i^f} \sum_{v_r \in t_j^f} C(v_q, v_r) \quad C(v_q, v_r) = \begin{cases} 1 & \text{if } v_q = v_r \\ 0 & \text{otherwise} \end{cases}$$
    • $t^m$ and $t^s$ represent the feature vector of a node used in the matching function and the similarity function, respectively.
Relation Extraction – Our Experience

• An Attempt to Apply Kernel-based Approach
  – Entity Tree Kernel $K(T_1, T_2)$
    • Matching Features: $t^m = \{\text{entity type, relation-argument}\}$
    • Similarity Features: $t^s = \{\text{entity type, entity subtype, entity head character Uni-gram}\}$

\[
K(t_1, t_2) = \begin{cases} 
0 & l(c_1) = l(c_2) \\
\sum_{i=1} K(c_1[i], c_2[i]) + s(t_1, t_2) & \text{otherwise}
\end{cases}
\]

\[
K_m(t_1, t_2) = \begin{cases} 
0 & l(c_1) \neq l(c_2) \text{ or } m(t_1, t_2) = 0 \\
m(t_1, t_2) & l(c_1) = l(c_2) = 0 \\
\prod_{i=1} K_m(c_1[i], c_2[i]) & \text{otherwise}
\end{cases}
\]
Relation Extraction – Our Experience

• An Attempt to Apply Kernel-based Approach
  – Feature-based vs. Kernel-based
    • Features
      – $t^m, t^m$
      – *Position Features: Adjacent vs. Separated, and Nested vs. Non-nested

<table>
<thead>
<tr>
<th>Types/Sub-Types</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature-based</td>
<td>66.24</td>
<td>58.06</td>
<td>61.88</td>
</tr>
<tr>
<td>Kernel-based</td>
<td>84.35</td>
<td>60.49</td>
<td>70.46</td>
</tr>
</tbody>
</table>
Relation Extraction – Our Experience

• An Attempt to Apply Kernel-based Approach
  – Similarity Features

<table>
<thead>
<tr>
<th>Similarity Feature</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>50.22</td>
<td>32.53</td>
<td>39.49</td>
</tr>
<tr>
<td>Sub-type</td>
<td>74.80</td>
<td>55.63</td>
<td>63.81</td>
</tr>
<tr>
<td>Head Uni-Gram</td>
<td>86.00</td>
<td>51.31</td>
<td>64.27</td>
</tr>
<tr>
<td>Type + Sub-type</td>
<td>76.65</td>
<td>53.99</td>
<td>63.36</td>
</tr>
<tr>
<td>Type + Head Uni-Gram</td>
<td>81.91</td>
<td>57.88</td>
<td>67.83</td>
</tr>
<tr>
<td>Sub-type + Head Uni-Gram</td>
<td>85.26</td>
<td>59.66</td>
<td>70.20</td>
</tr>
<tr>
<td>Type + Sub-type + Head Uni-Gram</td>
<td>84.35</td>
<td>60.49</td>
<td>70.46</td>
</tr>
</tbody>
</table>
Relation Extraction – Our Experience

• An Attempt to Apply Kernel-based Approach
  – Head Characters vs. Head Words

<table>
<thead>
<tr>
<th>Similarity Feature</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uni-Gram</td>
<td>84.35</td>
<td>60.49</td>
<td>70.46</td>
</tr>
<tr>
<td>Bi-Gram</td>
<td>83.38</td>
<td>58.82</td>
<td>68.98</td>
</tr>
<tr>
<td>Uni-Gram + Bi-Gram</td>
<td>84.38</td>
<td>60.20</td>
<td>70.27</td>
</tr>
<tr>
<td>Segmented Word</td>
<td>83.15</td>
<td>58.97</td>
<td>69.00</td>
</tr>
</tbody>
</table>
Relation Extraction – Our Experience

• An Attempt to Apply Kernel-based Approach
  – Entity Content Scope vs. Entity Context

<table>
<thead>
<tr>
<th>Similarity Feature</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F-Measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Head</td>
<td>84.35</td>
<td>60.49</td>
<td>70.46</td>
</tr>
<tr>
<td>Extent</td>
<td>85.12</td>
<td>60.68</td>
<td>70.85</td>
</tr>
<tr>
<td>Head + Extent</td>
<td>85.03</td>
<td>60.64</td>
<td>70.79</td>
</tr>
<tr>
<td>Head + Context</td>
<td>84.91</td>
<td>64.16</td>
<td>73.09</td>
</tr>
<tr>
<td>Extent + Context</td>
<td>85.37</td>
<td>63.14</td>
<td>72.59</td>
</tr>
<tr>
<td>Head + Extent + Context</td>
<td>85.83</td>
<td>63.14</td>
<td>72.76</td>
</tr>
</tbody>
</table>
Joint Extraction Approach

• Joint Extraction of Entity and Relation
  – Joint Extraction of Entities and Relations for Opinion Recognition (EMNLP2’006)
Joint Extraction Approach

• Joint Extraction of Entity and Relation
  – 2 Types of Entities
    • Opinion Entity: expresses opinion
    • Source Entity: denotes source of opinion
  – Relation between Opinion Entity and Source Entity
  – Example

\[Bush^{(1)} \text{ intends}^{(1)} \text{ to curb the increase in harmful gas emissions and is counting on}^{(1)} \text{ the good will}^{(2)} \text{ of } [US \text{ industrialists}]^{(2)} .\]
Joint Extraction Approach

• Joint Extraction of Entity and Relation
  – Opinion and Source Entity Extraction
    • Learning Model: Independent CRF Models
    • Feature Set: Local Features such as Lexical and Syntactic Information
    • Output: $n$-Best Extracts
  – Relation Link Classification
    • Learning Model: ME Model, trained on all pairs of entity extracted from above entity extraction
    • Feature Set: Lexical and Syntactic Information of Entities and Connections (Dependency Path, Patterns)
Joint Extraction Approach

• Joint Extraction of Entity and Relation
  – Global Inference

• Inference Model: Integer Linear Programming (ILP) Problem

Objective function $f$

\[
\begin{align*}
&= \sum_i (w_{o_i} O_i) + \sum_i (\bar{w}_{o_i} \bar{O}_i) \\
&+ \sum_j (w_{s_j} S_j) + \sum_j (\bar{w}_{s_j} \bar{S}_j) \\
&+ \sum_{i,j} (w_{i,j} L_{i,j}) + \sum_i (\bar{w}_{i,j} \bar{L}_{i,j})
\end{align*}
\]

$O_i$ and $\bar{O}_i$ : Opinion Variables
$S_j$ and $\bar{S}_j$ : Sources Variables
$L_{i,j}$ and $\bar{L}_{i,j}$ : Relation Variables

\[
\begin{align*}
\text{s.t.} & \quad \forall i, \ O_i + \bar{O}_i = 1 \\
& \quad \forall j, \ S_j + \bar{S}_j = 1 \\
& \quad \forall i, j, \ L_{i,j} + \bar{L}_{i,j} = 1 \\
& \quad \forall i, \ O_i = \sum_j L_{i,j} \\
& \quad \forall j, \ S_j + A_j = \sum_i L_{i,j} \\
& \quad \forall j, \ A_j - S_j \leq 0 \\
& \quad \forall i, j, i < j, \ X_i + X_j = 1, X \in \{S, O\}
\end{align*}
\]
IE Applications and Directions

• Fact Extraction vs. Opinion Extraction
• Closed Domain vs. Open Domain Extraction
• Knowledge and Database Construction
  – Database Population, Terminology Extraction and Ontology Population/Evolving
  – Cross-Document and Cross/Document Extraction
• Integration in IR, QA and Summarization
• Commercial and Scientific Applications Applications
References

• Introduction to Information Extraction (by Douglas E. Appelt and David J. Israel)

• Information Extraction (by Sunita Sarawagi)
References

• Information Extraction (by Jim Cowie and Yorick Wilks)

• Information Extraction (by Ralph Grishman)