Concept Extraction Challenge: University of Twente at #MSM2013

Mena B. Habib, Maurice van Keulen, and Zhemin Zhu
Database Chair
Agenda

- Introduction.
- Named Entity Extraction:
  - SVM.
  - CRF.
  - Hybrid approach.
- Named Entity Categorization:
  - Named Entity Disambiguation.
  - Entity Categorization.
- Results.
- Conclusion.
Introduction

Name Entity Recognition = Name Entity Extraction + Name Entity Categorization

Name Entity Categorization = Name Entity Disambiguation + Entity Categorization

RT @breakingnews: Blast in South Island, New Zealand, coal mine leaves 30 miners unaccounted for - Reuters

LOC

LOC

.../wiki/South_Island

.../wiki/New_Zealand

.../wiki/Reuters
Named Entity Extraction

- **SVM:**
  - Use TwiNER (Li et al @ SIGIR 2012) approach for segmenting tweet.
  - Yago KB is also used to enrich the NE candidates to achieve high recall.
  - Some hypothesis are applied to improve precision (removing stop words & verbs)

- Different features are extracted for each segment to train and test the SVM (like POS, AIDA disambiguation score, MS Web-Ngram probability, Shape features, frequency, etc.)
Named Entity Extraction

- CRF:
  - CRF is popular for sequence labeling. But training of CRFs can be very expensive due to the global normalization (linear-chain CRFs):
    - quadratic in the size of the label set and almost quadratic in the size of the training sample
  - We used method called empirical training.
    - The maximum likelihood estimation (MLE) of the empirical training has a closed form solution, and it does not need iterative optimization and global normalization. (Fast!)
    - The MLE of the empirical training is also a MLE of the standard training. (Precise!)
  - Tweet text is tokenized. For each token, the following features are extracted and used to train the CRF:
    - The Part of Speech (POS) tag of the word.
    - The word shape.
Named Entity Extraction

- Hybrid approach:
  - We take the union of the CRF and SVM results, after removing duplicate extractions, to get the final set of annotations.
  - For overlapping extractions we select the entity that appears in Yago, then the one having longer length.
Named Entity Categorization

- Named Entity Disambiguation:
  - AIDA disambiguation system is used to disambiguated the extracted NE.
  - $\sim 75.8\%$ of training data NEs $\in$ YAGO KB.
  - For NEs $\notin$ YAGO, we look for the first token in the NE if it $\in$ YAGO, if found we pick the entity with the higher prior probability. (Ex: “Sara MacDonald” is assigned to “.../wiki/Sara_Sidle”)
  - Other NEs $\notin$ YAGO at all are assigned to --NME--.
Named Entity Categorization

- Entity Categorization:
  - We build a profile for each category (PER, LOC, ORG, and MISC) from the Wikipedia Categories of each disambiguated entity.
  - If \((NE \in \text{Training set})\) \(\rightarrow\) Use category with the highest prior probability;
  - Else if (NE assigned to an entity) \(\rightarrow\) Find the most similar category profile to the Wikipedia Categories of the disambiguated entity;
  - Else \(\rightarrow\) Assign NE to PER category; //used with 2.8% of the extracted entities.
Results

- 4-fold cross validation.

<table>
<thead>
<tr>
<th>Extraction Results</th>
<th>Pre.</th>
<th>Rec.</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twiner Seg.</td>
<td>0.0997</td>
<td>0.8095</td>
<td>0.1775</td>
</tr>
<tr>
<td>Yago</td>
<td>0.1489</td>
<td>0.7612</td>
<td>0.2490</td>
</tr>
<tr>
<td>Twiner∪Yago</td>
<td>0.0993</td>
<td>0.8139</td>
<td>0.1771</td>
</tr>
<tr>
<td>Filter(Twiner∪Yago)</td>
<td>0.2007</td>
<td>0.8066</td>
<td>0.3214</td>
</tr>
<tr>
<td>SVM</td>
<td>0.7959</td>
<td>0.5512</td>
<td>0.6514</td>
</tr>
<tr>
<td>CRF</td>
<td>0.7157</td>
<td>0.7634</td>
<td>0.7387</td>
</tr>
<tr>
<td>CRF∪SVM</td>
<td>0.7166</td>
<td>0.7988</td>
<td>0.7555</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Extraction and Classification Results</th>
<th>Pre.</th>
<th>Rec.</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRF</td>
<td>0.6440</td>
<td>0.6324</td>
<td>0.6381</td>
</tr>
<tr>
<td>AIDA Disambiguation + Entity Categorization</td>
<td>0.6545</td>
<td>0.7296</td>
<td>0.6900</td>
</tr>
</tbody>
</table>
Conclusion

- We split the NER task into two separate tasks:
  - NEE which aims only to detect entity mention boundaries in text.
  - NEC which assigns the extracted mention to its correct entity type.
- For NEE we used a hybrid approach of CRF and SVM to achieve better results.
- For NEC we used AIDA disambiguation system to disambiguate the extracted named entities and hence find their type.
Thank You
Cases where SVM extracts other NE than CRF

217: "_Mention_ : Joy ! **MS Office** now syncs with *Google Docs* -LRB- well , in beta anyway -RRB- . We are soon to be one big happy collaborative Click family . *Ric*

245: _Mention_ " ` valleylist " " v135 r. 1 - - electricity -LRB- jerry yang and david filo -RRB- <NEWLINE> _URL_ "

**Bold** $\rightarrow$ SVM

*Italic* $\rightarrow$ CRF