Improved Pattern Learning for Bootstrapped Entity Extraction

Sonal Gupta and Christopher Manning
Department of Computer Science
Stanford University
Positive  Negative  Unlabeled

- Closed-world assumption: Assume negative Ignore
- Predict their labels

- E.g. most distantly supervised relation extraction systems sample unlabeled examples as negative
  - Recent work has addressed the problem (Ritter et al., TACL 2013; Xu et al., ACL 2013)
Problem

Similar closed world problem in bootstrapped pattern-based learning systems:

Unlabeled text is either treated as negative or is ignored
Contribution: Improving Bootstrapped Pattern Scoring

Predict labels of unlabeled entities using unsupervised measures to score patterns in bootstrapped pattern-based entity extraction.
Bootstrapped entity extraction

Unlabeled Text

Diseases
asthma attack
diabetes
pain

Treatments
ibuprofen
surgery
pain meds

Diseases
high bp
shoulder injury
ACL tear
carpel tunnel

Treatments
advair
holy basil
turmeric
statins
Objective

• Entity extraction in specialized domains (e.g. biology, medicine, law) using very little supervision (seed sets of entities)
  – No fully labeled data
  – Little coverage in Wikipedia, WordNet, Freebase, ...
  – Not web scale - no list wrappers

• Bootstrapped pattern-based learning of entities
Background: Patterns

• Surface word patterns
  I live in X:NN, STATE  I live in Stanford, CA

• Interpretable, effective, and widely used in industrial systems  Chiticariu et al. 2013

• A part of hybrid systems like NELL, KnowItAll
Background: *Bootstrapped* pattern-based learning

- Seed set of entities and unlabeled text
- Label data using the current set of entities
- Create candidate patterns
- Score candidate entities and select top $n$
- Select top $k$ patterns and apply them
- Score candidate patterns
- Select top $k$ patterns and apply them

$T$ iterations

Thelen and Riloff, 2002

Gupta and Manning: Improved Pattern Learning
**Bootstrapped pattern-based learning**

Seed set of entities and unlabeled text → Label data using the current set of entities

Entities belonging one entity type are positive and all other types negative.
Example

Learning ‘animal’ entities

*I own a **dog**, Tommy. I run with my pet **dog** and nap with my pet cat. I also own a house.*

Seed set: {dog}
Bootstrapped pattern-based learning

1. Seed set of entities and unlabeled text
2. Label data using the current set of entities
3. Create candidate patterns
Example

Learning ‘animal’ entities

*I own a dog*, Tommy. *I run with my pet dog and nap with my pet cat. I also own a house.*

Seed set: {dog}

Candidate patterns:

*own a X*
Positive: {dog}, Unlabeled: {house}

*my pet X*
Positive: {dog}, Unlabeled: {cat}
Bootstrapped pattern-based learning

1. Seed set of entities and unlabeled text
2. Label data using the current set of entities
3. Create candidate patterns
4. Score candidate patterns
Learning ‘animal’ entities

I own a dog, Tommy. I run with my pet dog and nap with my pet cat. I also own a house.

Seed set: {dog}

Candidate patterns:

- own a X
  - Positive: {dog}, Unlabeled: {house} Score: $s_1$

- my pet X
  - Positive: {dog}, Unlabeled: {cat} Score: $s_2$

If $s_2 > s_1$
Bootstrapped pattern-based learning

Seed set of entities and unlabeled text

Label data using the current set of entities

Score candidate entities and select top n

Create candidate patterns

Select top k patterns and apply them

Score candidate patterns

T iterations

Gupta and Manning: Improved Pattern Learning...
Example

Learning ‘animal’ entities

I own a dog, Tommy. I run with my pet dog and nap with my pet cat. I also own a house.

Seed set: {dog}  Learned entities: {cat}

Candidate patterns:

- own a X

- my pet X
Bootstrapped pattern-based learning

Seed set of entities and unlabeled text

Label data using the current set of entities

Score candidate entities and select top n

Create candidate patterns

Select top k patterns and apply them

Score candidate patterns

T iterations

Gupta and Manning: Improved Pattern Learning...
Pattern Scoring: If fully supervised...

• **If** we had labeled data,
  
  own a $\mathbf{X}$ $<$ my pet $\mathbf{X}$
  
  positive: \{dog\} positive: \{dog, cat\}
  
  negative: \{house\}

But, we don’t have labeled data to score patterns
Past Work

• Past work makes closed world assumption

• Unlabeled entities are either
  – Assumed negative: too conservative (Carlson et al., 2010)
    \[
    \text{own a } X = \text{my pet } X
    \]
  – Ignored (Downey et al., 2004)
    does not differentiate good vs bad unlabeled entities
    \[
    \text{own a } X = \text{my pet } X
    \]
  – Both (Yangarber et al. 2002, Lin et al. 2003)
Past Work: Pattern Scoring

One commonly used measure is $\text{RlogF}$ (Riloff, 1996 and Thelen and Riloff, 2002)

Pattern Score:

$$\frac{\#\text{pos}}{\#\text{pos} + \#\text{neg} + \#\text{unlabeled}} \log(\#\text{pos})$$

None of the previous work predicts labels of unlabeled entities.
Solution

• If we can exploit unsupervised sources
  E.g.
  \[
  \text{similarity(cat, dog)} > \text{similarity(house, dog)}
  \]

own a \(X\) < my pet \(X\)

positive: \{\text{dog}\} positive: \{\text{dog}\}

negative: \{\text{house: 0.8}\} negative: \{\text{cat: 0.1}\}

Gupta and Manning: Improved Pattern Learning...
Exploit **unsupervised** scoring measures to **evaluate unlabeled** entities to score patterns more accurately
Pattern Scoring Function

- Pattern score depends on entities it extracts
  - Positive and negative entities
  - Unlabeled entities

\[
\frac{\text{num of positive entities}}{\text{expected num of negative entities}} \log(\text{num positive})
\]
\[
\text{num of negative} + \sum_{e \in \text{unlabeled}} \text{score}(e \in \text{negative})
\]
Evaluating unlabeled entities

• $\text{score}(e \in \text{negative})$
  – between 0 and 1
  – We use five unsupervised weak predictors and average their scores

  – Learning a logistic regression classifier to combine features gave lower performance
    • Sampled unlabeled as negative!
Predictors for an unlabeled entity

• Misspelling or variation of already known entities
  
  – Feature 1: Edit distance from positive entities
    • E.g. ‘pinacillin’ for ‘penicillin’ for the type ‘Treatment’
  
  – Feature 2: Edit distance from negative entities
Predictors for an unlabeled entity

• In a specialized domain, more likely negative if commonly occurs in generic text (e.g. ‘youtube’)

  – Feature 3: Google Ngram TF-IDF score
Predictors for an unlabeled entity

• Close to positive or negative entities in word vector space

  – Feature 4: cluster entities using distributional similarity and predict probability of the label of the clusters
Predictors for an unlabeled entity

• Substring of multi-word positive vs negative seed phrases (e.g. more likely to see ‘John’ in NAME than in PLACE phrases)

  – Feature 5: ratio of frequency of the phrase in positive vs negative phrases
Experiments

• Extract ‘drug-and-treatment’ entities from MedHelp online health forums starting with a seed set

• Examples:

  I plan to start *cinnamon* and *holy basil* – known to lower glucose in many people.

  My sinus infections were treated electrically, with *high voltage million volt electricity*, which solved the problem, but the *treatment* is not FDA approved and generally unavailable, except under experimental treatment protocols.
Experiments

Metrics: Precision and Recall

- Precision: % correct among extracted
- Stop learning if precision drops below 75%

- Recall: % correct entities among the total unique correct entities pooled from all systems (maintaining minimum 75% precision)
Results

ACNE  65k sentences

[Graph showing precision-recall curves for different systems, including OurSystem, RlogF-PUN, Yangarber02, SqrtAllRatio, Lin03, and PUNOdd.]
Results

DIABETES 63k sentences
Results

ASTHMA  39k sentences

Graph showing the precision and recall for different systems, including 'OurSystem', 'RlogF-PUN', 'Yangarber02', 'SqrtAllRatio', 'Lin03', and 'PUNOdd'. The x-axis represents recall (out of 221 correct entities) ranging from 0.1 to 1, while the y-axis represents precision ranging from 0.76 to 0.96.
I am put on **Cortisone** by the doctor.

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Positive</th>
<th>Negative</th>
<th>Unlabeled</th>
<th>Our system Rank</th>
<th>Best Baseline</th>
</tr>
</thead>
<tbody>
<tr>
<td>i be put on X</td>
<td>cortisone, prednisone, asmanex, advair, augmentin, ...</td>
<td>inhaler, inhalers, hfa</td>
<td>8</td>
<td>Not extracted</td>
<td></td>
</tr>
<tr>
<td>he give I more X</td>
<td>antibiotics, steroid, antibiotic</td>
<td>pinacillin</td>
<td>68</td>
<td>Not extracted</td>
<td></td>
</tr>
</tbody>
</table>

**He gave me more steroids.**

Close in word vector space

Small edit distance from positive seed entity “penicillin”
Conclusion

• Existing bootstrapped pattern-based learning systems make closed world assumptions
  – Unlabeled entities are either ignored or considered negative

• Predicting labels of unlabeled entities when scoring patterns significantly improves entity extraction

• Models that contrast domain-specific and general text, and use distributional similarity and edit distance measures are useful

• Future: apply to relation extraction, distantly-supervised learning, and other semi-supervised approaches.
### Advertisement:
In Workshop on Interactive Language Learning, Visualization, and Interfaces

<table>
<thead>
<tr>
<th>6. rhinoquart</th>
<th>6. anitboitics</th>
</tr>
</thead>
<tbody>
<tr>
<td>7. seritide</td>
<td>7. samuterol</td>
</tr>
<tr>
<td>8. flixotide</td>
<td>8. nexum</td>
</tr>
<tr>
<td>9. omnacorticol</td>
<td>10. flexitide</td>
</tr>
<tr>
<td>11. methochline</td>
<td>Iteration 2</td>
</tr>
<tr>
<td>12. advaer</td>
<td>13. asanex</td>
</tr>
<tr>
<td>14. porair</td>
<td>15. xeponex</td>
</tr>
<tr>
<td>16. levelabuterol</td>
<td></td>
</tr>
</tbody>
</table>

Gupta and Manning: Improved Pattern Learning...
Thanks!

• Download bootstrapped pattern-based learning code (part of Stanford CoreNLP v3.4)

• Download pattern visualization and diagnostics code:


sonal@cs.stanford.edu