Hearst Patterns Revisited:
Automatic Hypernym Detection from Large Text Corpora

Stephen Roller, Douwe Kiela, and Maximilian Nickel
**Hypernymy**

- Hierarchical relations play a central role in knowledge representation (Miller, 1995)
  
  cat is a feline is a mammal is an animal

  All animals are living things -> cats are living things

- Automatic hypernymy detection approaches:
  
  - **Pattern based**: high-precision lexico-syntactic patterns (Hearst, 1992)
  
  - **Distributional Inclusion**: unconstrained word co-occurrences (Zhitomirsky-Geffet and Dagan, 2005)

/ [NP] such as [NP] (and [NP])?/ animals such as cats and dogs animals including cats and dogs cats, dogs, and other animals
Objectives

- Are Hearst patterns more valuable than distributional information?
  - Do we learn more from using **general semantic contexts**, or exploiting **highly targeted ones**?
  - Are differences robust across multiple evaluation settings?

- Can we remedy some of Hearst patterns’ weaknesses?
  - Scaling up data and extraction is cheaper and easier today
  - Do embedding methods help alleviate sparsity?
Tasks

10% Validation, 90% Test

Detection

• Distinguish hypernymy pairs from other relations
• Average Precision (AP) across 5 datasets (Shwartz et al., 2017)

Direction

• Identify the direction of entailment (X ⇒ Y or Y ⇒ X?)
• Accuracy across 3 datasets (Kiela et al., 2015)
• 2 also contain non-entailments (X ⇔ Y)

Graded Entailment

• Predict the degree of entailment
• Spearman’s rho on 1 dataset (Vulić et al., 2017)

Detection

• BLESS (Baroni and Lenci, 2011)
• EVAL (Santus et al., 2015)
• LEDS (Baroni et al., 2012)
• Shwartz (Shwartz et al., 2016)
• WBLESS (Weeds et al., 2014)

Direction

• BLESS (Baroni and Lenci, 2011)
• WBLESS (Weeds et al., 2014)
• BiBless (Kiela et al., 2015)

Graded Entailment

• Hyperlex (Vulić et al., 2017)
Hearst Pattern Extraction

Preprocessing

• 10 Hearst patterns
• Gigaword + Wikipedia
  • Lemmatized, POS tagged
• Matches were aggregated and filtered:
  • Pair must match 2 distinct patterns
• 431K distinct pairs covering 243K unique types

<table>
<thead>
<tr>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>X which is a (example</td>
</tr>
<tr>
<td>X (and</td>
</tr>
<tr>
<td>X which is called Y</td>
</tr>
<tr>
<td>X is JJS (most)? Y</td>
</tr>
<tr>
<td>X a special case of Y</td>
</tr>
<tr>
<td>X is an Y that</td>
</tr>
<tr>
<td>X is a !(member</td>
</tr>
<tr>
<td>!(features</td>
</tr>
<tr>
<td>(Unlike</td>
</tr>
<tr>
<td>Y including X₁, X₂, ...</td>
</tr>
</tbody>
</table>
Hearst Pattern Models

Count transformation

- **PPMI**(x, y): transform counts using Positive Pointwise Mutual Information

Simple embedding (Truncated SVD)

- **SPMI**(x, y): apply truncated SVD to PPMI counts
- Select k using validation set
- Related to Cederberg and Widdows (2003)
Distributional Methods

- Cosine baseline
- Selected 3 high performing, unsupervised methods based on Shwartz et al. (2017)
  - WeedsPrec (Weeds et al., 2004); invCL (Lenci and Benotto, 2012); SLQS (Santus et al., 2014)
- Use strong distributional space from Shwartz et al. (2017)
  - Wikipedia + UkWaC
  - POS tagged and lemmatized
  - Dependency contexts (Pado and Lapata, 2007; Levy and Goldberg, 2014)
- Tune hyperparameters on validation
Detection

- Distr. methods have trouble with global calibration (AP)
- Pattern has mixed performance
- SPMI model best on 4/5 datasets.
- Embedding Hearst patterns helps overcome sparsity
  - Fills in gaps
  - Downweights outliers

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Cosine</th>
<th>Best Distributional</th>
<th>PPMI</th>
<th>SPMI</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLESS</td>
<td>.12</td>
<td>.19</td>
<td>.45</td>
<td>.76</td>
</tr>
<tr>
<td>Shwartz</td>
<td>.31</td>
<td>.28</td>
<td>.43</td>
<td>.44</td>
</tr>
<tr>
<td>EVAL</td>
<td>.29</td>
<td>.36</td>
<td>.39</td>
<td>.48</td>
</tr>
<tr>
<td>LEDS</td>
<td>.69</td>
<td>.72</td>
<td>.69</td>
<td>.72</td>
</tr>
<tr>
<td>WBLESS</td>
<td>.96</td>
<td>.84</td>
<td>.84</td>
<td>.71</td>
</tr>
</tbody>
</table>
Detection + Direction difficult for distributional methods

Patterns outperform distr. methods on 2/3

BLESS pathologically difficult for cosine and PPMI

SPMI significantly better

Embedding patterns overcome sparsity

**Direction**

![Accuracy Chart]

- **Cosine**
- **Best Distributional**
- **PPMI**
- **SPMI**
- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

<table>
<thead>
<tr>
<th>Hyperlex</th>
<th>Spearman’s rho</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cosine</td>
<td>0.14</td>
</tr>
<tr>
<td>Best Distributional</td>
<td>0.43</td>
</tr>
<tr>
<td>PPMI</td>
<td>0.60</td>
</tr>
<tr>
<td>SPMI</td>
<td>0.53</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hyperlex</th>
<th>Spearman’s rho</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cosine</td>
<td>0.14</td>
</tr>
<tr>
<td>Best Distributional</td>
<td>0.50</td>
</tr>
<tr>
<td>PPMI</td>
<td>0.60</td>
</tr>
<tr>
<td>SPMI</td>
<td>0.53</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hyperlex</th>
<th>Spearman’s rho</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cosine</td>
<td>0.14</td>
</tr>
<tr>
<td>Best Distributional</td>
<td>0.43</td>
</tr>
<tr>
<td>PPMI</td>
<td>0.60</td>
</tr>
<tr>
<td>SPMI</td>
<td>0.53</td>
</tr>
</tbody>
</table>

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment

Graded Entailment

- Pattern based methods outperform distr.
- Embedding hurts...
  - Spearman’s rho doesn’t punish ties (many 0s)
  - Add small noise ($10^{-6}$) to PPMI model to break ties randomly
- SPMI best after adjustment
Conclusions

• Pattern-based approaches outperform distributional methods
  • Targeted Hearst contexts are more valuable than semantic similarity gains

• Embedding Hearst patterns works well
  • Helps substantially with sparsity issues

• We open source our experiments and evaluation framework:
  https://github.com/facebookresearch/hypernymysuite
Thank you!

Questions?