Lexical entailment
Some foundations
What is hyponymy?

- A structuring relation of vocabulary

- For instance fruit $\leftrightarrow$ apple
  
  - Apple: hyponym of fruit
  
  - Fruit: superordinate (hyperonym) of apple
Hyponymy – two Difficulties

• Hyponymy often defined as entailment between sentences

• BUT
  • Does not invariably entail the corresponding sentence with the hyperonym

• I.e.:

  'It’s not a tulip.' ≠ 'It’s not a flower'

  'That it was a tulip surprised her.' ≠ 'That it was a flower surprised her.'
Entailment $\iff$ hyponymy

• **Entailments:**
  • Need to be context independent
    • i.e. „Not all dogs are pets.“ but in everyday life „Dogs are pets.“

• **Hyponymy:**
  • Context sensitive
Hyponymy in speech

',X is a type / kind / sort of Y‘

• A horse is a type of animal.

• * A kitty is a sort of cat.

• > A kitty is a young cat
Hyponymy, a transitive relation?

If A is a hyponym of B, and B is a hyponym of C, then A is C′
Hyponymy, a transitive relation? – Examples

A car-seat is a type of seat.
A seat is a type of furniture.

* A car-seat is a type of furniture.
Hypernymy Detection
Why hypernymy?

Which actors are involved in scientology?
Another example

• The bow lute, such as the Bambara ndang, is plucked and has an individual curved neck for each string.

NP0 such as \{NP1, NP2, ..., (and\mid or)\} NPn

• Approach to pattern-based interpretation techniques
How to find constructions

• Occur frequently and in many text genres
• Almost always indicate the relation of interest
• Can be recognized with little or no pre-encoded knowledge
Vered Shwartz, Yoav Goldberg, Ido Dagan

Improving Hypernymy Detection
Determining hypernymy

• Automated methods to determine for given term-pairs \((x,y)\)
  • Is \(y\) an hypernym of \(x\)?

• Two approaches
  • Distributional
  • Path-based
Distributional method

• Distributional representation of the term-pair
• Decision based on separate context of x and y
• No requirement of occurring together
  - Less precise detecting specific semantic similarity between the terms
Path-based method

• Deciding on lexico-syntactic paths/patterns connecting the joint occurrences of x and y
  • i.e. Y such as X

• Individual paths as features result in huge, sparse feature space
  • Some patterns are rare
    • „spelt is a species of wheat“
      • X be species of Y
    • „Fantasy is a genre of fiction“
      • X be genre of Y
    • Both indicate X is-a Y
HypeNET

• Integration of a path-based and distributional method
• Uses a long short-term memory (LSTM) network (neural network) to encode dependency paths
  • Training data constructed on knowledge resources used by other models
    • for comparing reason
LSTM-based Hypernymy Detection
HypeNETs Path-based Network – Edge Representation

• Represent each dependency path as a sequence of edges that lead from x to y in the dependency tree

• Each edge contains
  • The lemma
  • POS tag
  • Dependency label
  • Edge direction

\[ \vec{u}_e = [\vec{u}_l, \vec{u}_{pos}, \vec{u}_{dep}, \vec{u}_{dir}] \]
HypeNETs Path-based Network – Path Representation

• Path $p$ composed of
  • Edges $e_1, \ldots, e_k$
  • Edge vectors $v_{e_1}, \ldots, v_{e_k}$

• Fed in order to an LSTM encoder
  • Resulting in a vector $\overrightarrow{o_p}$

• LSTM
  • Effective at capturing temporal pattern in sequences
Term-Pair Classification

• Each (x,y) term-pair is represented by
  • Multiset of lexico-syntactic paths
  • Representation of each (x,y) term-pair as a weighted-average of its path vectors

\[
\mathbf{v}_{xy} = \mathbf{v}_{\text{paths}(x,y)} = \frac{\sum_{p \in \text{paths}(x,y)} f_p(x,y) \cdot \mathbf{v}_p}{\sum_{p \in \text{paths}(x,y)} f_p(x,y)}
\]

(1)

• \(f_p(x,y)\): frequency of \(p\) in \(\text{paths}(x,y)\)
• Path vector fed to a single-layer network (binary classification)

\[
c = \text{softmax}(W \cdot \mathbf{v}_{xy})
\]

(2)
Term-Pair Classification
Implementation details

• Train the network with PyCNN (a neural network library)
• Minimize cross entropy with gradient-based optimization
  • Mini-batches of size of 10
  • Adam update rule
• Initialized the lemma embeddings with
  • Pre-trained GloVe word embeddings (trained on Wikipedia)
  • Tried 50- and 100-dimensional embedding vectors
    • Selected the better performing
  • Other embeddings and out-of-vocabulary lemmas randomly initialized
Dataset
Creating Instances

• Need large amounts of data

• Problem:
  • Hypernymy datasets are relatively small

• Solution:
  • Used distant supervision from knowledge resources
    • WordNet
    • Wikidata
    • Yago
  • Prevent including questionable relations
    • Denoting positive examples per hand selected

• Ratio of 1:4 positive to negative pairs
Random and Lexical Dataset Splits

• Primary dataset
  • Standard random splitting
    • 70% train
    • 25% test
    • 5% validation sets

• Problem with supervised distributional lexical inference methods
  • Tend to perform ”lexical memorization“
    • i.e.
      (dog, animal), (cat, animal), (cow, animal) -> (x, animal) => (paper, animal)

<table>
<thead>
<tr>
<th></th>
<th>train</th>
<th>test</th>
<th>val</th>
<th>all</th>
</tr>
</thead>
<tbody>
<tr>
<td>random split</td>
<td>49,475</td>
<td>17,670</td>
<td>3,534</td>
<td>70,679</td>
</tr>
<tr>
<td>lexical split</td>
<td>20,335</td>
<td>6,610</td>
<td>1,350</td>
<td>28,295</td>
</tr>
</tbody>
</table>
Baseline
Baseline – Path-based method

Snow

• Extracted all shortest paths of four edges or less between terms in a dependency tree
• Add paths with „satellite edges“
  • i.e. such Y as X
• Number of distinct paths was 324,578
  • Apply $\chi^2$ feature selection to keep only 100,00 most informative paths
Baseline – Path-based method Generalizion

• Compared their method to a baseline with generalized dependency paths
  • Replaced edges with their POS tags and wildcards
• Generated the powerset of all possible generalizations

<table>
<thead>
<tr>
<th>path</th>
</tr>
</thead>
<tbody>
<tr>
<td>X/NOUN/dobj/&gt; establish/VERB/ROOT/- as/ADP/prep/&lt; Y/NOUN/pobj/&lt;</td>
</tr>
<tr>
<td>X/NOUN/dobj/&gt;  VERB as/ADP/prep/&lt; Y/NOUN/pobj/&lt;</td>
</tr>
<tr>
<td>X/NOUN/dobj/&gt;  * as/ADP/prep/&lt; Y/NOUN/pobj/&lt;</td>
</tr>
<tr>
<td>X/NOUN/dobj/&gt; establish/VERB/ROOT/- ADP Y/NOUN/pobj/&lt;</td>
</tr>
<tr>
<td>X/NOUN/dobj/&gt; establish/VERB/ROOT/- * Y/NOUN/pobj/&lt;</td>
</tr>
</tbody>
</table>

• Number of features went up to 2,093,220
  • Kept only 1,000,000
Baseline – distributional methods unsupervised

• SLQS: entropy-based measure for hypernymy detection
• Applied the vanilla setting of SLQS on their dataset
  • Bad result, due to containing rare items
  • Improved later with changed settings and items

• Validation set used for the beginning to tune for
  • the classification of pairs as positive
  • The maximum number of each terms most associated contexts (N)

• SLQS performs better for classifying specificity level of related terms than hypernymy
Baseline – distributional methods supervised

• Represent term-pairs with distributional features with 3 state-of-the-art methods
  • Concatenation
  • Difference
  • Dot-product

• Used several pre-trained embeddings of different sizes
  • Trained on three classifiers
    • Logistic regression
    • SVM
    • SVM with RBF kernel
Results
## Results

<table>
<thead>
<tr>
<th>Method</th>
<th>random split</th>
<th></th>
<th></th>
<th>lexical split</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>precision</td>
<td>recall</td>
<td>$F_1$</td>
<td>precision</td>
<td>recall</td>
<td>$F_1$</td>
</tr>
<tr>
<td>Path-based</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Snow</td>
<td>0.843</td>
<td>0.452</td>
<td>0.589</td>
<td>0.760</td>
<td>0.438</td>
<td>0.556</td>
</tr>
<tr>
<td>Snow + Gen</td>
<td>0.852</td>
<td>0.561</td>
<td>0.676</td>
<td>0.759</td>
<td>0.530</td>
<td>0.624</td>
</tr>
<tr>
<td>HypeNET Path-based</td>
<td>0.811</td>
<td>0.716</td>
<td>0.761</td>
<td>0.691</td>
<td></td>
<td>0.632</td>
</tr>
<tr>
<td>Distributional</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLQS (Santus et al., 2014)</td>
<td>0.491</td>
<td>0.737</td>
<td>0.589</td>
<td>0.375</td>
<td>0.610</td>
<td>0.464</td>
</tr>
<tr>
<td>Best supervised (concatenation)</td>
<td>0.901</td>
<td>0.637</td>
<td>0.746</td>
<td>0.754</td>
<td>0.551</td>
<td>0.637</td>
</tr>
<tr>
<td>Combined</td>
<td>HypeNET Integrated</td>
<td>0.913</td>
<td>0.890</td>
<td>0.901</td>
<td>0.809</td>
<td>0.617</td>
</tr>
</tbody>
</table>
Analysis
Qualitative Analysis

• HypeNET finds high-scoring paths of true-positives
  • In the path-based baseline, these are the highest-weighted features
  • LSTM less straightforward at identifying the most indicative paths
    • Success because of considering a certain path $p$ as the only path for an appeared term-pair and compute it as a TRUE label score

\[
\text{softmax}(W \cdot v_{xy})[1], \text{ setting } v_{xy} = [\vec{0}, \vec{\phi}_p, \vec{0}]
\]
Qualitative Analysis - Snow

• Snows method can only rely on verbatim paths
  • > limitation of its recall

• Its generalized version leads to coarse generalization
  • i.e. X VERB Y from
    • X take Y from
    • Avoidance in generalization lead to lower recall

• HypeNET provides a better midpoint
  • Fine-grained generalization by learning additional example paths
Further notice – Random split

• HypeNET learned a range of specific paths
  • i.e. as X is Y for e.g. Y=magazine
  • X is Y produced e.g. Y= film

• HypeNET noticed ‘X is Y’
  • is a “noisy” path
    • E.g. (chocolate, problem) for
      „Chocolate is a big problem in the context of children’s health.”
Error Analysis – False positives

• Occurred for random splits
  • To sum up semantic relations they used board categories
    • E.g. synonym, includes alias and Wikipedia redirections

• 20% stem from confusing synonymy with hypernymy
  (but are known to be difficult to distinguish)

• 30% were reversed term-pairs, hypernym-hyponym
  • Closer examination revealed pairs of near-synonyms
    • i.e. not clear whether one term more general than the other
      • Fiction in WordNet hypernym of story
      • Fiction in their classification as hyponym

• Some were hypernym-like relations

• Other correspond to rare term-pairs
Error Analysis – False negatives

• Most pairs with few co-occurrences in the corpus
  • i.e. (night, play) for „Night“, a dramatic sketch by Harold Pinter

• Such term-pairs with too few hypernymy-indicating paths
Conclusion
HypeNET

• First, focus on improving path representation using LSTM
• Next, extend network by integrating distributional signals

• Architecture seems straightforwardly applicable, could be used for semantic relations
Questions?