Olive Oil is Made of Olives, Baby Oil is Made for Babies
Interpreting Noun Compounds using Paraphrases in a Neural Model

Vered Shwartz\(^1\) and Chris Waterson\(^2\)
\(^1\) Computer Science Department, Bar-Ilan University.  \(^2\) Google Inc.

1. Noun Compound Classification

**Task Definition:** Given a noun compound \(w_1w_2\), classify the relation between \(w_1\) and \(w_2\) to one of a set of pre-defined relations.

<table>
<thead>
<tr>
<th>SUBSTANCE</th>
<th>PART OF</th>
<th>PURPOSE</th>
<th>NON-COMPOSITIONAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>olive oil</td>
<td>dog</td>
<td>game</td>
<td>hot dog</td>
</tr>
<tr>
<td>apple cake</td>
<td>dog</td>
<td>room</td>
<td>horse radish</td>
</tr>
<tr>
<td>leather jacket</td>
<td>dog</td>
<td>service</td>
<td></td>
</tr>
<tr>
<td>missile battery</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Best Baseline:** \(\text{qual} = 0.600\), \(\text{sem} = 0.645\), \(\text{int} = 0.478\), \(\text{low} = 0.538\)

2. Distributional Approaches

- Compute a vector for \(w_1w_2\) as a function of \(w_1\) and \(w_2\)’s vectors
  - \(\text{vec(oil olive)} = f(\text{vec(oil)}, \text{vec(olive)})\), many ways to learn \(f\) (e.g. \([1, 2]\))
- Use noun-compound representation as a feature vector for classification
  - \([3]\): Best performance is achieved when \(f(w_1, w_2) = [w_1; w_2]\) 😊
  - There is a lexical memorization issue \([4]\)

3. Using Paraphrases for Classification

**Paraphrase Embeddings:**

- **Motivation (paraphrases):** describe the relation between \(w_1\) and \(w_2\) explicitly: \([w_2]\) extracted from \([w_1]\)
- **Motivation (embeddings):** semantic generalization of paraphrases: \([w_2]\) obtained from \([w_1]\)
- **Input:** dependency paths connecting \(w_1\) and \(w_2\) in a corpus: “oil made of olives”, “oil extracted from olives”, etc.
- **Learn paraphrase representation using HypeNET \([5]\):**
- **Objective:** predict relation from path embedding

**Classification Models:**

- MLP
- \(\text{path}([w_1; w_2])\)
- \(\text{mean pooling}\)
- \([w_1]; [w_2]\)
- \([w_1]; \text{made of } [w_2]\)
- \([w_1]; \text{from } [w_2]\)
- \([w_2]\) extracted from \([w_1]\)

4. Evaluation Settings

- **Dataset:** Tratz-2011 \([6]\): 19,158 instances
  - 37 relations (fine-grained) / 12 relations (coarse-grained)
- **Splits:** (1) Random, (2) Lexical-full, (3) Lexical-head, (4) Lexical-mod

5. Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Split</th>
<th>Best Baseline</th>
<th>Path</th>
<th>Int</th>
<th>Int-NC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tratz-fine</td>
<td>Random</td>
<td>0.725</td>
<td>0.538</td>
<td>0.714</td>
<td>0.692</td>
</tr>
<tr>
<td></td>
<td>Lex-head</td>
<td>0.458</td>
<td>0.448</td>
<td>0.510</td>
<td>0.478</td>
</tr>
<tr>
<td></td>
<td>Lex-mod</td>
<td>0.667</td>
<td>0.472</td>
<td>0.613</td>
<td>0.600</td>
</tr>
<tr>
<td></td>
<td>Lex-full</td>
<td>0.363</td>
<td>0.423</td>
<td>0.421</td>
<td>0.429</td>
</tr>
<tr>
<td>Tratz-coarse</td>
<td>Random</td>
<td>0.775</td>
<td>0.588</td>
<td>0.736</td>
<td>0.712</td>
</tr>
<tr>
<td></td>
<td>Lex-head</td>
<td>0.538</td>
<td>0.518</td>
<td>0.558</td>
<td>0.548</td>
</tr>
<tr>
<td></td>
<td>Lex-mod</td>
<td>0.645</td>
<td>0.548</td>
<td>0.646</td>
<td>0.632</td>
</tr>
<tr>
<td></td>
<td>Lex-full</td>
<td>0.409</td>
<td>0.472</td>
<td>0.475</td>
<td>0.478</td>
</tr>
</tbody>
</table>

Random split: baseline performs best Paraphrases contribute in lexical splits

6. Relations Expressed in Paraphrases

- **Measure:** \([w_2]\) varies by \([w_1]\)
- **Supply:** \([w_2]\) manufacture \([w_1]\)
- **Time:** \([w_2]\) held Saturday \([w_1]\)
- **Substance:** \([w_2]\) made of wood and \([w_1]\)

**Non-Compositional:** soap opera, bullet train

7. Looking into the Compound Embeddings

- **Inf-NC and Dist-NC aren’t usually better than Inf and Dist. Why?**
  - 1) No embeddings? Most noun-compounds (81%) have embeddings
  - 2) Low quality embeddings? Semantically-similar noun-compounds have similar embeddings
  - 3) Inconsistent annotations: majority party, equative, minority party, whole+part_or_member_of, enforce: director, engagement chief, perform:engage_in

8. References


9. Contact

Code: https://goo.gl/yI2FgM. Questions: vered1986@gmail.com