Aspect Term Extraction for Sentiment Analysis: New Datasets, New Evaluation Measures, and an Improved Unsupervised Method

John Pavlopoulos and Ion Androutsopoulos
NLP Group, Department of Informatics
Athens University of Economics and Business, Greece
http://nlp.cs.aueb.gr/
Aspect Based Sentiment Analysis

- User generated reviews
- Aspect terms carry sentiment
- Aspect Based Sentiment Analysis:
  - Aspect Term Extraction
  - Sentiment Estimation per aspect term
  - Aspect Aggregation
- We focus on Aspect Term Extraction
## Previous datasets vs. our datasets

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Inter- Annotator Agreement</th>
<th># Domains</th>
<th>Gold Aspect Terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hu &amp; Liu 2004</td>
<td>✗</td>
<td>1</td>
<td>✓</td>
</tr>
<tr>
<td>Ganu et al. 2009</td>
<td>✓</td>
<td>1</td>
<td>✗</td>
</tr>
<tr>
<td>Blitzer et al. 2007</td>
<td>✗</td>
<td>4</td>
<td>✗</td>
</tr>
<tr>
<td>Pavlopoulos &amp; Androutsopoulos 2014</td>
<td>✓</td>
<td>3</td>
<td>✓</td>
</tr>
</tbody>
</table>
Our new datasets

<table>
<thead>
<tr>
<th>Domain</th>
<th>( n = 0 )</th>
<th>( n &gt; 0 )</th>
<th>( n &gt; 1 )</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restaurants</td>
<td>1,590</td>
<td>2,120</td>
<td>872</td>
<td>3,710</td>
</tr>
<tr>
<td>Hotels</td>
<td>1,622</td>
<td>1,978</td>
<td>652</td>
<td>3,600</td>
</tr>
<tr>
<td>Laptops</td>
<td>1,760</td>
<td>1,325</td>
<td>416</td>
<td>3,085</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Domain</th>
<th>( n &gt; 0 )</th>
<th>( n &gt; 1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restaurants</td>
<td>multi-word</td>
<td>single-word</td>
</tr>
<tr>
<td></td>
<td>593</td>
<td>452</td>
</tr>
<tr>
<td>Hotels</td>
<td>199</td>
<td>262</td>
</tr>
<tr>
<td>Laptops</td>
<td>350</td>
<td>289</td>
</tr>
</tbody>
</table>

Inter-Annotation Agreement
- Dice: \(~70\%\) in all domains
- Cohen’s Kappa not applicable
Multi-word vs. single-word distinct aspect terms per domain

Electronics (Hu & Liu, 2004) & Laptops reviews contain more multi-word distinct aspect terms!
Precision, Recall, F-measure

Gold:
“design” (94), “service” (3), “screen” (2)

Predicted:
“design” (92/94), “service” (1/3 + 1), “screen” (0/2), “foo” (+3)

Computed on types (distinct aspect terms):

\[
P = \frac{\text{true predicted types}}{\text{predicted types}} = \frac{2}{3} = 0.66
\]

\[
R = \frac{\text{true predicted types}}{\text{true types}} = \frac{2}{3} = 0.66
\]

Computed on tokens (aspect term occurrences):

\[
P = \frac{\text{true predicted tokens}}{\text{predicted tokens}} = \frac{92+1+0+0}{92+2+0+3} = 0.96
\]

\[
R = \frac{\text{true predicted tokens}}{\text{true tokens}} = \frac{92+1+0+0}{94+3+2+0} = 0.94
\]
What we want to measure

• The users care only about the top $m$ (e.g., 10-20) most frequently discussed distinct aspect terms.
  • The value of $m$ depends on screen size, available time etc.

• Among the $m$ truly most frequent distinct aspect terms, finding or missing a truly more frequent distinct aspect term should be rewarded or penalized more.

• Among the $m$ distinct aspect terms returned by a method, placing a truly high-frequency distinct aspect term towards the beginning of the returned list should be rewarded more.
How we propose to measure

- Order all the correct distinct aspect terms by human annotation frequency ($G$ list).
- Each method returns a list of distinct aspect terms, ordered by predicted frequency ($A$ list).
  - Given an $m$ value, use the first $m$ elements of the $A$ list ($A_m$).
- Compare $G$ and $A_m$ for different $m$ values.
Weighted Precision, Recall

\[ WP_m = \frac{\sum_{i=1}^{m} \frac{1}{i} \cdot 1 \cdot \{a_i \in G\} }{\sum_{i=1}^{m} \frac{1}{i}} \quad m=3 \quad \frac{1}{1} + 0 + \frac{1}{3} = 0.73 \]

\[ WR_m = \frac{\sum_{i=1}^{m} \frac{1}{r(a_i)} \cdot 1 \cdot \{a_i \in G\} }{\sum_{j=1}^{\mid G \mid} \frac{1}{j}} \quad m=3 \quad \frac{1}{1} + \frac{1}{2} + \frac{1}{3} = 0.82 \]

- By varying \( m \), we obtain \( WP_m - WR_m \) curves.
- Also, average weighted precision at 11 weighted recall levels.
- \( WP_m \) is similar to \( nDCG@m \), but no counter-part for \( WR_m \).
Methods

Freq baseline

- Effective and popular **unsupervised** baseline (Liu, 2012)
- Returns the **most frequent nouns and noun phrases**, ordered by decreasing sentence frequency

H&L (Hu & Liu 2004)

- Also **unsupervised**, finds frequent nouns and noun phrases, plus...
- Discards candidate aspect terms that are **subparts of** other candidate aspect terms
- Finds **adjectives that modify candidate aspect terms**, uses them to detect **additional candidate aspect terms** (nouns, noun phrases modified by the adjectives)
- Details previously unclear, **full pseudo-code** in our paper
Word2Vec-based pruning step

Applicable to both *Freq* and *H&L*

We use word vectors (Mikolov, 2013) computed using *Word2Vec*

\[ \mathbf{v}('king') - \mathbf{v}('man') + \mathbf{v}('woman') \sim \mathbf{v}('queen') \]

\[ \mathbf{v}('queen'):\text{ word vector of ‘queen’} \]
\[ <0.2, 0.9, 0.0, \ldots, 0.3, 0.7, 0.5> \]

\[ \mathbf{v}_{domain} = \frac{\sum_{w \in \text{aspect terms}} \mathbf{v}(w)}{|\text{aspect terms}|} \]

\[ \mathbf{v}_{common} = \frac{\sum_{w \in \text{common words}} \mathbf{v}(w)}{|\text{common words}|} \]

For each candidate aspect term \( \mathbf{a} \), measure its similarity (cosine) with the two centroids

Prune \( \mathbf{a} \), if \( \cos(\mathbf{a}, \mathbf{v}_{aspect}) < \cos(\mathbf{a}, \mathbf{v}_{common}) \)
Results: avg weighted precision

<table>
<thead>
<tr>
<th>Method</th>
<th>Restaurants</th>
<th>Hotels</th>
<th>Laptops</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freq</td>
<td>43.40</td>
<td>30.11</td>
<td>9.09</td>
</tr>
<tr>
<td>Freq+w2v pruning</td>
<td>45.17</td>
<td>30.54</td>
<td>7.18</td>
</tr>
<tr>
<td>Hu&amp;Liu</td>
<td>52.23</td>
<td>49.73</td>
<td>34.34</td>
</tr>
<tr>
<td>H&amp;L+w2v pruning</td>
<td>66.80</td>
<td>53.37</td>
<td>38.93</td>
</tr>
</tbody>
</table>

All differences are statistically significant (p<0.01) – Approximate Stratified Randomization
Results: weighted precision-recall

![Graph showing the results for weighted precision-recall for different models: Freq, Hu & Liu, Hu & Liu + W2V, and Freq + W2V. The graph plots weighted precision on the y-axis against weighted recall on the x-axis.](image-url)
Results: weighted precision-recall

![Graph showing weighted precision and recall for hotels with different methods: Freq, Hu & Liu, Hu & Liu + W2V, Freq + W2V.](image)
Results: weighted precision-recall
Conclusions

- 3 new aspect term extraction datasets
  - Laptops/Restaurants/Hotels
  - Available upon request
- New evaluation measures
  - Weighted precision, weighted recall, average weighted precision
- Improved the popular unsupervised method of Hu & Liu
  - Additional pruning step based on continuous space word vectors (using Word2Vec)
Thank you!

Questions?