Evaluating Lexical Substitution: Analysis and New Measures

Sanaz Jabbari, Mark Hepple, Louise Guthrie

Department of Computer Science
University of Sheffield
Overview

- Lexical Substitution
- SemEval–2007: English Lexical Substitution Task
- Metrics: analysis and revised metrics
  - Notational Conventions
  - Best Answer Measures
  - Measures of Coverage
  - Measures of Ranking
Lexical Substitution

- **Lexical Substitution Task (LS):**
  - find replacement for target word in sentence, so as to preserve meaning (as closely as possible)
  - e.g. replace target word *match* in: *They lost the match*
  - possible substitute: *game* — gives: *They lost the game*

- Target words may be *sense ambiguous*
  - so, task implicitly requires *word sense disambiguation* (WSD)
  - in above e.g., context disambiguates target *match*, and so determines what may be good substitutes

- McCarthy (2002) proposed LS be used to *evaluate WSD systems*
  - implicitly requires WSD
  - approach side-steps divisive issues of standard WSD evaluation
  - e.g. what is the appropriate *sense inventory*?
The English Lexical Substitution Task (ELS07):
- task at SemEval–2007

Test items = sentence with an identified target word
- systems must suggest substitution candidates

Items selected to be targets were:
- all sense ambiguous
- ranged over parts-of-speech (N, V, Adj, Adv)
- ~200 targets terms, 10 test sentences each

Gold standard:
- 5 annotators, asked to propose 1–3 substitutes per test item
- gold standard records set of proposed candidates
- and the count of annotators that proposed each candidate
  - assumed that a higher count indicates a better candidate
Notational Conventions

- Test data consists of $N$ items $i$, with $1 \leq i \leq N$
- Let $A_i$ denote system response for item $i$ (answer set)
- Let $H_i$ denote human proposed substitutes for item $i$ (gold std)
- Let $freq_i$ be a function returning the count for each term in $H_i$
  - i.e. count of annotators proposing that term
  - for any term not in $H_i$, $freq_i$ returns 0
- Let $maxfreq_i$ denote maximal count of any term in $H_i$
- Let $m_i$ denote the mode answer for $i$
  - exists only if item has a single most-frequent response
For any set of terms $S$, use $|S|^i$ to denote the *summed count values* of the terms in $S$ according to $freq_i$, i.e.:

$$|S|^i = \sum_{a \in S} freq_i(a)$$

**EXAMPLE:**

Assume item $i$ with target *happy* (adj), with human answers:

- $H_i = \{ \text{glad, merry, sunny, jovial, cheerful} \}$
- and associated counts: $(3, 3, 2, 1, 1)$
- abbreviate as: $H_i = \{ G:3, M:3, S:2, J:1, Ch:1 \}$

**THEN:**

- $maxfreq_i = 3$
- $|H_i|^i = 10$
- mode $m_i$ is *not defined* ($> 1$ terms share max value)
Best Answer Measures

• Two ELS07 tasks involve finding a ‘best’ substitute for test item

FIRST TASK: system can return set of answers $A_i$. Score as:

$$\text{best}(i) = \frac{|A_i|^i}{|H_i|^i \times |A_i|}$$

◇ have $|A_i|^i$ above: summed ‘count credits’ for answer terms
◇ have $|A_i|$ below: number of answer terms

• so returning $> 1$ term only allows system to ‘hedge its bets’
• optimal answer includes only a single term having max count value

• PROBLEM:

◇ dividing by $|H_i|$ means even optimal response gets score well below 1

Example: for gold std example $H_i = \{G:3, M:3, S:2, J:1, Ch:1\}$

optimal answer set $A_i = \{G\}$ gets score $\frac{3}{10}$ or 0.3
Problem fixed by removing $|H_i|$, and dividing instead by $\text{maxfreq}_i$:

$$(\text{new}) \ best(i) = \frac{|A_i|^i}{\text{maxfreq}_i \times |A_i|}$$

**EXAMPLES:** with gold std $H_i = \{G:3,M:3,S:2,J:1,Ch:1\}$, find:

- optimal answer $A_i = \{G\}$ gets score 1
- good 'hedged' answer $A_i = \{G,S\}$ gets score 0.83
- hedged good/bad answer $A_i = \{G,X\}$ gets score 0.5
- weak but correct answer $A_i = \{J\}$ gets score 0.33
Best Answer Measures (contd)

• **SECOND TASK**: requires single answer from system
  ◊ its ‘best guess’ answer \( bg_i \)
  ◊ answer receives credit only if it is *mode answer* for test item:

\[
mode(i) = \begin{cases} 
1 & \text{if } bg_i = m_i \\
0 & \text{otherwise}
\end{cases}
\]

• **PROBLEMS:**
  ◊ reasonable to have task where only single term allowed
  ◊ *BUT* has some key limitations — approach:
    • is *brittle* — only applies to items with a unique mode
    • * loses information* valuable to ranking systems
      i.e. *no credit* for answer that is good but not mode
Best Answer Measures  (contd)

• Instead, propose *should* have a ‘single answer’ task
  ◇ *BUT* don’t require a *mode* answer
  ◇ *rather*, assign full credit for an *optimal answer*
  ◇ *but* lesser credit also for a correct/non-optimal answer

• Metric — the *best-1* metric:

\[
best_1(i) = \frac{freq_i(bg_i)}{maxfreq_i}
\]

i.e. score 1 if \(freq_i(bg_i) = maxfreq_i\)

◇ lesser credit for answers with lower human count values

◇ metric applies to all test items
Measures of Coverage

- Third ELS07 task: 'out of ten' (oot) task
  - tests if systems can field a wider set of substitutes
  - systems may offer set $A_i$ of up to 10 guesses
  - metric assesses proportion of total gold std credit covered

$$\text{oot}(i) = \frac{|A_i|^i}{|H_i|^i}$$

- PROBLEM: does nothing to penalise incorrect answers

- ALTERNATIVE VIEW: if aim is to return a broad set of answer terms
  - an ideal system will return all and only the correct substitutes
  - a good system will return as many correct answers as possible, and as few incorrect answers as possible
Measures of Coverage (contd)

- This view suggests instead what metrics like *precision* and *recall*
  - to *reward* correct answer terms (recall), and
  - to *punish* incorrect ones (precision)
  - taking *count weightings* into account

- Definitions *without* count weighting (not the final metrics):
  - *correct answer terms* given by:  $|H_i \cap A_i|$
  - Recall:
    $$R(i) = \frac{|H_i \cap A_i|}{|H_i|}$$
  - Precision:
    $$P(i) = \frac{|H_i \cap A_i|}{|A_i|}$$
Measures of Coverage  (contd)

- For the *weighted* metrics, no need to intersect $H_i \cap A_i$
  - count function $freq_i$ assigns count 0 to incorrect terms
  - so *weighted correct terms* is just $|A_i|^i$

- Recall (weighted):
  \[
  R(i) = \frac{|A_i|^i}{|H_i|^i}
  \]
  - same as *oot* metric (but no limit to 10 terms)

- For precision — issue arises:
  - what is the ’count weighting’ of *incorrect* answers?
  - must specify a *penalty* factor — applied per incorrect term

- Precision (weighted):
  \[
  P(i) = \frac{|A_i|^i}{|A_i|^i + k|A_i - H_i|}
  \]
Measures of Coverage (contd)

- EXAMPLES:
  - Assume same gold std $H_i = \{G:3,M:3,S:2,J:1,Ch:1\}$
  - Assume penalty factor $k = 1$
  - Answer set $A_i = \{G, M, S, J, Ch\}$
    - all and only the correct terms
    - gets $P = 1, R = 1$
  - Answer set $A_i = \{G, M, S, J, Ch, X, Y, Z, V, W\}$
    - contains all correct answers plus 5 incorrect ones
    - gets $R = 1$, but only $P = 0.66 \ (10/(10 + 5))$
  - Answer set $A_i = \{G, S, J, X, Y\}$
    - has 3 out of 5 correct answers, plus 2 incorrect ones
    - gets $R = 0.6 \ (6/10)$ and $P = 0.75 \ (6/6 + 2)$
Measures of Ranking

- Argue that *core* task for LS is *coverage*

- Coverage tasks will mostly be tackled by combining:
  - method to *rank* candidate terms (drawn from lexical resources)
  - means of drawing a *boundary* between good ones and bad

- So, may be useful to have means to assess *ranking* ability *directly*
  
  i.e. to aid process of system development

- Method (informal):
  - consider *list* of up to 10 candidates from system
  - at each rank position 1..10, compute what (count-weighted) proportion of *optimal* performance an answer list achieves
  - average over the 10 values so-computed
Measures of Ranking (contd)

\[ H_i = \{ G:3, M:3, S:2, J:1, Ch:1 \} \mapsto \]

<table>
<thead>
<tr>
<th>rank</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>freq</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cum.freq</td>
<td>3</td>
<td>6</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

\[ A_i = (S, Ch, M, J, G, X, Y, Z, V) \mapsto \]

<table>
<thead>
<tr>
<th>rank</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>freq</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cum.freq</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

\[ rank(i) = \frac{1}{10} \times (\frac{2}{3} + \frac{3}{6} + \frac{6}{8} + \frac{7}{9} + \frac{10}{10} + \frac{10}{10} + \frac{10}{10} + \frac{10}{10} + \frac{10}{10}) = 0.87 \]
Measures of Ranking (contd)

\[ H_i = \{ G:3, M:3, S:2, J:1, Ch:1 \} \mapsto \]

<table>
<thead>
<tr>
<th>rank</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>freq</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cum.freq</td>
<td>3</td>
<td>6</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

\[ A_i = (X, Y, S, Ch, M, Z, J, V, G) \mapsto \]

<table>
<thead>
<tr>
<th>rank</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>freq</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>cum.freq</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>3</td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

\[ rank(i) = \frac{1}{10} \times \left( \frac{0}{3} + \frac{0}{6} + \frac{2}{8} + \frac{3}{9} + \frac{6}{10} + \frac{6}{10} + \frac{7}{10} + \frac{7}{10} + \frac{10}{10} + \frac{10}{10} \right) = 0.52 \]