English Lexical Substitution Task for SemEval

Diana McCarthy and Roberto Navigli

January 8, 2007

1 Introduction

This document contains the information about scoring for the LEXSUB task at SEMEVAL. It includes information on the format of the input files, the gold standard files and the format required for the system output files. It also contains the details needed for running the scorer and the measures used for evaluation.

There are three types of scoring. Systems can be evaluated on any subset of these scoring types:

best Scoring the best substitutes for a given item

oot Scoring for the best 10 substitutes for a given item. 10 responses are anticipated and systems will not benefit from providing less responses [1]

mw precision and recall for detection and identification of multiwords in the input sentences

The details of scoring for these types are described below in section 4. First we will describe the format of the input files, the gold standard files and the system output files.

2 Format

Please note that in this section we are using { } brackets to indicate variables in our textual description so that we can distinguish variables from xml tags. The variables used in our equations in section 4 will be indicated with symbols introduced in the text of section 4.

[1] We would like to thank Suzanne Stevenson for suggesting this option.

2.1 Input Format for Trial and Test Set: see trial dataset lexsub_trial.xml

The file input to systems for evaluation will adhere to the following format:

```
<corpus lang="english">
   <lexelt item="{lemma}.{pos}">
      <instance id="{id}">
         <context>...<head>...</head>...</context>
      </instance>

      ⋮

     <instance id="{id}">
         <context>...<head>...</head>...</context>
      </instance>
   </lexelt>

   ⋮

   <lexelt item="{lemma}.{pos}">
      <instance id="{id}">
         <context>...<head>...</head>...</context>
      </instance>

      ⋮

      <instance id="{id}">
         <context>...<head>...</head>...</context>
      </instance>
   </lexelt>
</corpus>
```

where each `<lexelt>` tag focuses on a specific lemma and part of speech, as specified in the attribute `item`. {`pos`} can assume one of the following four values: `a`, `v`, `n`, `r` (for adjective, verb, noun, adverb, respectively). Alternatively, it can be specified in the format {`ori_pos`}.{`pos`} where {`ori_pos`} is the original part of speech (PoS) automatically produced from the PoS tagger and {`pos`} is the manually corrected PoS. An example of a corrected PoS is at id 131 in the trial data where *stand* was tagged as a noun in the PoS tagged corpus, but from the annotators responses it was apparent that it was functioning as a verb [2] so the lexelt was changed from *stand.n* to *stand.n.v*.

[2]The annotators are not given the PoS of the target word.

Each sentence is represented with an <instance> tag (each specifying a unique numeric id attribute. Each <instance> tag contains a <context> tag which includes the sentence in which an instance of the lemma co-occurs. The word instance is in turn enclosed in a <head> tag. For instance:

```
<corpus lang="english">

   ⋮

   <lexelt item="bright.a">
      <instance id="3">
         <context>The roses have grown out of control ,
         wild and carefree , their <head>bright</head>
         blooming faces turned to bathe in the early
         autumn sun .</context>
      </instance>
   </lexelt>

   ⋮

</corpus>
```

2.2 Gold Standard and System Output Formats

The gold-standard format will be the same for the **best** and **oot** evaluation. The **mw** gold-standard will have a different format. The system output files will differ for all 3 scoring methods.

Please note that all human responses are semi-automatically lemmatised so systems should ensure that all their answers are provided in lemmatised form.

Please note that if the humans have used a hyphen (-) in a response then we will accept a space instead of a hyphen from the system output as correct. If a system uses a hyphen, but not the annotators then the system will be marked wrong.

2.2.1 gold standard format for best and oot: see example file gold.trial

This file is provided by the task organisers. The format is

`{lexelt}\s{id}\s::\s{list of substitues with frequency}`

where the `\s` represents a single space and `lexelt` is the `lemma.pos` (or `lemma.ori_pos.pos`) described in the section 2.1 above. Each item of the list or substitutes is separated by ; and consists of the lemmatised word of phrase and a frequency count indicating the number of annotators that provided this substitute.

Example:
bright.a 1 :: intelligent 3;clever 2;smart 1;
bright.a 2 :: light 2;luminous 2;clear 1;

2.2.2 system format for best: see example file BL.out

The file output by systems for evaluation should confirm to the format:

`{lexelt}\s{id}\s::{list of substitutes}` where the `\s` represents a single space and `lexelt` is the `lemma.pos` (or `lemma.ori_pos.pos`) described in the section 2.1 above. Each item of the list of substitutes is separated by ; and consists of the lemmatised word or phrase. The best guess should appear first in the list. Example:
severely.r 127 :: seriously
tight.r 32 :: fast
wild.a.n 160 :: natural state;state of nature

The order of ids in the file does not matter.

In case the results file contains two or more lines for the same reference number for the same reference id, the first such line will be counted as the system's answer and the subsequent lines will be disregarded.

2.2.3 system format for oot: see example file BLoutof10.out

The file output by systems for evaluation should confirm to the format:

`{lexelt}\s{id}\s:::\s{list of substitutes}`

(N.B. three colons to differentiate from the **best** output file!)

where the `\s` represents a single space and `lexelt` is the `lemma.pos` (or `lemma.ori_pos.pos`) described in the section 2.1 above. Each item of the list of substitutes is separated by ; and consists of the lemmatised word or phrase. Systems can provide up to 10 substitutes and will not have any advantage by providing less
Example:
wild.a 151 ::: mad;excited;frantic;chaotic;frenzied;manic;disorderly;unrestrained;delirious;unsubdued
manage.v 95 ::: negociate;bring off;pull off;carry off

The order of ids in the file does not matter.

In case the results file contains two or more lines for the same reference number for the same reference id, the first such line will be counted as the system's answer and the subsequent lines will be disregarded.

2.2.4 gold standard format for mw: see example file mwgold.trial

This file is provided by the task organisers. The format is:

`{lexelt}\s{id}\s::\s{list of identified multiwords with frequency}`

where the `\s` represents a single space and `lexelt` is the `lemma.pos` (or `lemma.ori_pos.pos`) described in the section 2.1 above. Each item of the list of multiwords identified is separated by ; and consists of the lemmatised multiword phrase and a frequency count indicating the number of annotators that identified this multiword. Example:
take.v 29 :: take place 5;
cross.n 59 :: cross section 4;

2.2.5 system format for mw: see example file dummyMW.out

The file output by systems for evaluation should confirm to the format:

`{lexelt}\s{id}\s:: multiword`

where the `\s` represents a single space and `lexelt` is the `lemma.pos` (or `lemma.ori_pos.pos`) described in the section 2.1 above. `multiword` is the identified multiword with component words separated by spaces.
Example:
cross.n 54 :: cross section

In case the results file contains two or more lines for the same reference number for the same reference id, the first such line will be counted as the system's answer and the subsequent lines will be disregarded.

3 Running the Scorer: score.pl

The scorer is a perl program. To run this do:

`perl score.pl system_file gold_file  [-t best|oot|mw] [-v]`

where

required parameters	
system_file	is the file of formatted answers output by a system.
gold_file	is the file with the gold standard provided by human annotators.

optional parameters	
-t	specifies the type of scoring: best, oot (out of ten) or mw (multiword) with best as the default.
-v	causes line-by-line scoring calculations to be printed.

For example:

```
perl score.pl  BL.out gold.trial
perl score.pl  BL.out gold.trial -v
perl score.pl BLoutof10.out gold.trial -t oot
perl score.pl BLoutof10.out gold.trial -t oot -v
perl score.pl dummyMW.out mwgold.trial -t mw
perl score.pl dummyMW.out mwgold.trial -t mw -v
```

4 Details of the Evaluation Measures

We have 3 separate scoring functions to allow scoring on

1. any number of best guesses, with best first
2. up to 10 guesses (no penalising for multiple guesses to cope with fact that we only have 5 annotators and systems may come up with a larger, but equally valid, set of substitutes)
3. multiword detection (spotting that the target is part of a multiword) and multiword identification (specifying the actual multiword)

Let H be the set of annotators, T be the set of test items with 2 or more responses (non NIL or proper name) from the annotators and h_i be the set of responses for an item $i \in T$ for annotator $h \in H$.

For each $i \in T$ we will calculate the mode (m_i) which is the most frequent response, provided that there is a response more frequent than the others. The set of items where there is such a mode is referred to as TM. Let A (and AM) be the set of items from T (or TM) where the system provides at least one substitute. Let $a_i : i \in A$ (or $a_i : i \in AM$) be the set of guesses from the system for item i. For each i we calculate the multiset union (H_i) for all h_i for all $h \in H$ and for each unique type (res) in H_i will have an associated frequency ($freq_{res}$) for the number of times it appears in H_i. For example: Given an item (id 9999) for ***happy;a*** supposing the annotators had supplied answers as follows:

annotator	responses
1	glad merry
2	glad
3	cheerful glad
4	merry
5	jovial

and the system's responses for this item was *glad; cheerful* then H_i would be *glad glad glad merry merry cheerful jovial*. The res with associated frequencies would be *glad 3 merry 2 cheerful 1* and *jovial 1*.

4.1 Measures for best

This requires the **best** file produced by the system which gives as many guesses as the system believes are fitting, but where the credit for each correct guess will be divided by the number of guesses. The first guess in the list will be taken as the best.

$$precision = \frac{\sum_{a_i:i \in A} \frac{\frac{\sum_{res \in a_i} freq_{res}}{|a_i|}}{|H_i|}}{|A|} \tag{1}$$

$$recall = \frac{\sum_{a_i:i \in T} \frac{\frac{\sum_{res \in a_i} freq_{res}}{|a_i|}}{|H_i|}}{|T|} \tag{2}$$

$$Mode\ precision = \frac{\sum_{best\ guess_i \in AM} 1\ if\ best\ guess = m_i}{|AM|} \tag{3}$$

$$Mode\ recall = \frac{\sum_{best\ guess_i \in TM} 1\ if\ best\ guess = m_i}{|TM|} \tag{4}$$

Using the example for *happy;a* id 9999 in section 4, the credit for a_{9999} in the numerator of precision and recall would be $\frac{\frac{3+1}{2}}{7} = .286$

4.2 Measures for oot

This allows a system to make up to 10 guesses. The credit for each correct guess will not be divided by the number of guesses. There is no ordering of the guesses

$$precision = \frac{\sum_{a_i:i \in A} \frac{\sum_{res \in a_i} freq_{res}}{|H_i|}}{|A|} \tag{5}$$

$$recall = \frac{\sum_{a_i:i \in T} \frac{\sum_{res \in a_i} freq_{res}}{|H_i|}}{|T|} \tag{6}$$

$$Mode\ precision = \frac{\sum_{a_i:i \in AM} 1\ if\ any\ guess \in a_i = m_i}{|AM|} \tag{7}$$

$$Mode\ recall = \frac{\sum_{a_i:i \in TM} 1\ if\ any\ guess \in a_i = m_i}{|TM|} \tag{8}$$

4.3 Measures for mw

This allows a system to identify items where the target is part of a multiword and what the multiword is. The annotators do not all have linguistics background so the decision on whether a word is a multiword will depend on their gut feel (we will be releasing the instructions given to annotators). Both humans and systems are asked to give 1 response as to the multiword in the original sentence. Let MW be the subset of T for which there is a multiword response which more than one annotator has provided. Let $mw_i \in MW$ be the most frequent multiword from the humans. Let $MWsys$ be the subset of T for which there is a multiword response from the system and $mwsys_i$ be a multiword specified by the system for item i.

$$detection\ precision = \frac{\sum_{mwsys_i \in MWsys} 1\ if\ mw_i\ exists\ at\ i}{|MWsys|} \tag{9}$$

$$detection\ recall = \frac{\sum_{mwsys_i \in MW} 1\ if\ mw_i\ exists\ at\ i}{|MW|} \tag{10}$$

$$identification\ precision = \frac{\sum_{mwsys_i \in MWsys} 1\ if\ mwsys_i = mw_i}{|MWsys|} \tag{11}$$

$$identification\ recall = \frac{\sum_{mwsys_i \in MW} 1\ if\ mwsys_i = mw_i}{|MW|} \tag{12}$$

5 Baselines

We will continue to work on the baselines. Currently we have a baseline for **best** and **oot** tasks. These are provided with the output files (BL.out and BLoutof10.out). BL.out was produced by using the first listed synset in WordNet 2.1 for the target word and taking the synonym with the largest frequency (according to frequency data collected from grammatical relations obtained from the BNC). If there was no synonyms for the first listed synset then we used synonyms from the hypernyms (verbs and nouns) or closely related classes (adjectives) of that first synset and used the same frequency data for ranking these related words. We do not have frequency data for multiwords.

BLoutof10.out was produced by finding for each word up to 10 synonyms from the fist listed synset (ranked by the frequency lists used for BL.out) from WordNet 2.1 and then, filling any spare capacity with hypernyms (verbs and nouns) or closely related classes (adjectives) related to the first listed synset and subject to the same frequency ranking.

6 Measuring Human Agreement

We will measure pairwise agreement using the multisets. For each paired set of responses (h_i) from $i \in T$ for 2 annotators ($h \in H$) where both have provided a response, we calculate agreement as the multiset intersection divided by the multiset union. The sum of these pairwise scores is divided by the sum of all non nil paired annotator sets (h_i for all $h \in H$ and $i \in T$).

We also plan to look at individual annotator agreement with the mode for each item.

For multiwords we calculate the pairwise agreement of multiword identification over all items (many of these won't be multiwords so agreement will be high due to this), and for the subset of multiword responses we plan to calculate the pairwise agreement of the actual multiword which is identified.