**Event Mention Detection scoring**

The scorer reads the output of event mention detection systems and compares them to the gold standard.

**Input:**

1. Gold standard annotation for documents, in format (one line per mention)

2. System output annotation for documents, in format (one line per mention)

3. Tokenization files associated with each document

**Output:**

1. System output annotation for the same text, with score for each mention appended to each line

2. System’s overall performance report as described in “Scoring” section

**System and gold standard annotation file format:**

1. Event mention annotations of all documents are written into one single file.
2. First line of each document is a header.
   1. Header := #BeginOfDocument<s><doc ID>
3. Last line of each document is a footer
   1. Footer := #EndOfDocument

**Definition of event mention format:**

event-mention := <system ID><TAB><doc ID><TAB><token ID list><TAB> <mention><TAB><event-type><TAB><realis status><TAB><score1><TAB> <score2><TAB><score3>

**Explanation of above-mentioned notations:**

<engine ID> := the name of the system

<text ID> := the ID of the text

<token ID list> := List of token Id in ascending order, separated by comma (,)

<mention> := the actual character string of the mention

<event-type> := the ACE hierarchy type

<realis-status> := the REALIS label

<score1> := any score (confidence, etc.) the system wants to assign (ignored)

<score2> := score assigned in the evaluation

<score3> := additional possible score assigned by human

<s> := represent a space character

<TAB> := represent a tab character

**Scoring:**

Let mappingScores = {}

#compute overlap scores pairwisely

FOR each system mention S := {S\_mid, S\_tokens}

Let S\_mid := mention id of S

Let S\_tokens := token indices associated with S

Let S\_tokens := S\_tokens – {token indices of invisible words} #See NOTE 1

FOR each gold mention G:= {G\_mid, G\_tokens}

Let G\_mid := mention id of G

Let G\_tokens := token indices associated with G

Let G\_tokens := G\_tokens – {token indices of invisible words}

Let overlap := OVERLAP(S\_tokens, G\_tokens)

IF overlap > 0

mappingScores := mappingScores + (G, S, overlap)

END IF

END FOR

END FOR

#Find mapping based on overlap scores

Sort mappingScores based on overlap

Mapping = {}

WHILE mappingScores != {}:

(G, S, overlap) = mappingScores.pop()

IF G has not been mapped and S has not been mapped THEN Mapping := Mapping + {G,S, overlap}

END WHILE

#Append score to compare

FOR each gold mention G:= {G\_mid, G\_tokens}:

Score := Mapping[G].overlap

append Score to the end of the line of G\_mid in the Gold Standard file, in position <score2>

END FOR

#Compute overall performance

TP = 0

FN = 0

FOR EACH Gold Mention G

IF G contains in Mapping

TP := TP + Mapping[G].overlap

ELSE

FN := FN + 1

END IF

END FOR

#Performance based on precision, recall:

Precision := TP / (TP+FP)

Recall := TP / #GoldStandardMentions

Subroutine OVERLAP(G,S):

IF G == S, THEN score := 1.0

IF G∧S == {}, THEN score := 0.0

ELSE

IF |G| > |S|, THEN score := (|S∧G|)/|G|

IF |G| < |S|, THEN score := (|S∧G|)/|S|

End Subroutine

Note 1: Invisible words are ignored in scoring. They include: determiners, …?

Examples:

Rule 1: do not accept prepositions but include particles

* "[look] up a chimney" vs "[look up] a dictionary"
* "[climb] up the ladder"
* [take responsibility for]
* sing [all the way] to school
* [go] to school

Rule 2: consider the maximum extent of an event mention, but don't worry about determiners (they are invisible)

* [takes a shower] ==> it is okay for annotators to include "a" in their annotation; we can ignore "a" in evaluation
* [make a quick decision] ==> it is okay for annotators to annotate the whole phrase; we can ignore "a" and include "quick" in evaluation