

# Supplementary Material: Joint Event Trigger Identification and Event Coreference Resolution with Structured Perceptron

Jun Araki and Teruko Mitamura

Language Technologies Institute

Carnegie Mellon University

Pittsburgh, PA 15213, USA

{junaraki,teruko}@cs.cmu.edu

## 1 Structured Perceptron

We show the structured perceptron algorithm in Algorithm A. In our event graph learning, a training example is represented as a pair of input document  $x$  and its associated event graph  $y$ . Line 4 involves decoding to generate the best event graph for  $x$ .  $\Phi(x, y)$  denotes a feature vector function that computes a feature vector for event graph  $y$  over  $x$ .

---

### Algorithm A Structured perceptron.

---

**Input:** training examples  $\{(x^{(i)}, y^{(i)})\}_{i=1}^N$

**Input:** number of iterations  $T$

**Output:** weight vector  $\mathbf{w}$

```
1:  $\mathbf{w} \leftarrow \mathbf{0}$  ▷ Initialization.
2: for  $t \leftarrow 1..T$  do
3:   for  $i \leftarrow 1..N$  do
4:      $\hat{y}^{(i)} = \operatorname{argmax}_{y \in \mathcal{Y}(x^{(i)})} \mathbf{w} \cdot \Phi(x^{(i)}, y)$ 
5:     if  $\hat{y}^{(i)} \neq y^{(i)}$  then
6:        $\mathbf{w} \leftarrow \mathbf{w} + \Phi(x^{(i)}, y^{(i)}) - \Phi(x^{(i)}, \hat{y}^{(i)})$ 
7: return  $\mathbf{w}$ 
```

---

## 2 Features for Event Trigger Identification

We use the following features for event trigger identification:

1. Whether to end with a suffix ‘-tion’ or ‘-ment’
2. Whether to begin with a determiner
3. The number of tokens in the trigger
4. Lemma of the head word
5. Lower-case string of the trigger
6. Part-of-speech tag of the head word
7. All part-of-speech tags of the trigger
8. Relation type of dependency heads
9. Relation type of dependency children
10. Semantic role types if the trigger is a predicate
11. Levin verb classes of the head word
12. Whether the trigger is a title of pages under category ‘Biological processes’ and related categories

## 3 Features for Event Coreference Resolution

For the best-first clustering in event coreference resolution, we use the following pairwise features given two event triggers:

1. Whether 3-character prefixes match
2. Whether Lower-case strings match
3. Whether Head word strings match
4. Whether Lemma of head words match
5. Whether the Nomlex nominalization of head words match
6. Whether the following trigger is an acronym of the preceding trigger
7. Whether the following trigger is in apposition to the preceding trigger
8. Part-of-speeches of head words
9. Dependency path between the triggers
10. Whether the following trigger has a determiner
11. Whether both triggers are predicates
12. Whether both triggers share a semantic role
13. Whether both triggers has an argument of the same string with the same semantic role
14. Levin verb classes which both head words belong to
15. Whether the preceding trigger is the first word in a document