Natural Language Processing
Lab 12: Semantic Role Labeling

---

1 Introduction

In this lab, we are going to do PropBank style semantic annotation and study inter-annotator agreement. We are going to focus on one polysemous verb, the verb *count*, which occurs exactly 10 times in the training set of the English web treebank. All files needed are available from `/local/kurs/nlp/semantics/`.

2 Study the lexicon

The lexical entry for the verb *count* can be found in the file `count.senses`. It is organized by (numbered) senses, with the first sense being `count.01`:

```
<roleset id="count.01" name="enumerate" vncls="-">
<roles>
  <role descr="counter" n="0" />
  <role descr="thing counted" n="1" />
</roles>

<example name="Count von Count">
<text>
    But this is a production designed and directed by Franco
    Zeffirelli and paid for by Sybil Harrington, who *trace* has no
    need *trace* to count her pennies, unlike Violetta, down to 20
    louis at the opera's end.
</text>
        <arg n="0">*trace* -&gt; *trace* -&gt;  who -&gt;  Sybil Harrington</arg>
        <rel>count</rel>
        <arg n="1">her pennies</arg>
</example>

<note>
</note>
</roleset>
```

This sense is nicknamed `enumerate` and has two roles: ARG0 = the counter; ARG1 = the thing(s) counted. In the example given, the ARG0 role is filled by a chain of coreferring NPs (some of which are empty traces) ending in `Sybil Harrington`, while the ARG1 role is filled by the NP `her pennies`. A simpler example would be:

```
[ARG0 Sybil Harrington] doesn't count.01 [ARG1 her pennies].
```

Study the rest of `count.senses` and then answer the following questions:

1. How many senses are there altogether?
2. Construct one example of your own for each sense.

3 Annotate some data

The file `count.instances` contains all the sentences in the training set of the English web treebank that contains the verb lemma `count`. (The file format is the CoNLL-X format that you previously used in the dependency parsing lab.) Your task is to annotate each sentence in the following way:

1. Annotate the verb with the relevant sense.
2. Annotate all words in an argument with the relevant argument label.

We will keep things simple and simply add the relevant label in the last column of the CoNLL format.[1] Thus, here is what the annotation of a hypothetical examle should look like:

---

[1]In general, it is much better to use a dedicated annotation tool to prevent typographical errors when editing the file.

```
1    We       we       PRON     PRON      _    2    nsubj    _    A0
2    count    count    VERB     VERB      _    0    root     _    count.03
3    on       on       ADP      ADP       _    4    case     _    A1
4    you      you      PRON     PRON      _    2    nmod     _    A1
5    .        .        PUNCT    PUNCT     _    0    punct    _    _
```

Note that your task is restricted to the annotation of predicate senses (like `count.03`) and core argument roles (numbered roles like `A0`, `A1` and `A2`). You do not need to annotate modifier roles like `AM-TMP` and `AM-LOC`. Note also that, for prepositional phrases like *on you*, you should include the preposition *on* in the span of the argument. This is different from the annotations in `count.senses`, where the span only includes the noun phrase part (*you*) and the preposition is treated as a feature of the argument (`f="on"`).

4   Compare your annotations

Team up with a classmate (or two) and compare your respective annotations. **NB:** Make sure to save a copy of your original annotation before starting to compare and discuss them. Answer the following questions:

1. What percentage of verb senses did you agree on?
2. What percentage of argument *roles* did you agree on?
3. What percentage of argument *spans* did you agree on?

After identifying all disagreements, your final task is to resolve them: create a file annotated for the whole group.