Natural Language Processing

Assignment 2: Advanced Text Processing

---

1 Introduction

This assignment involves material from lectures 5, 6 and 7. You should have watched the relevant videos, read the relevant chapters in the textbook and made a serious attempt at completing the relevant labs before you attempt this assignment. If you feel you have done that and still find the instructions unclear, you are welcome to email the course teachers and/or go to office hours to ask for help. The assignment is split into 2 sections, one about POS tagging, one about Lemmatisation. Each section is worth 10 points. If you are interested in receiving a VG grade, you may complete the *optional* exercise available to you, which is related to Hidden Markov Models. Note that the exercise must be completed in full in order to receive VG credit. We expect between half a page and a page for each section, except when stated otherwise. Please do not submit more than 5 pages overall. Your answers for each section should be self-contained.

2 POS-Tagging

- In Lab 5, you have tuned a tagger. Based on the best version of your tagger, you should perform a manual error analysis where you go through at least 5 sentences and comment on the errors made by the tagger. Are the mistagged words genuinely ambiguous? Why do you think they were mistagged? Is it possible that some of the words are mistagged in the gold standard? [ca 1/2 page].[1]
- What tagsets exist for your native language? List the ones you can find (spend not more than 15min on the search) and describe one of them in more detail.
- Is it necessary to tokenize text before tagging it? Please motivate your answer and give at least one example.
- In the HMM lab we have investigated key sequences and predicted words. What do these correspond to when using HMMs for POS-tagging? Please motivate your answer.

3 Lemmatisation

- In Lab 7, you have tuned a lemmatizer. Based on the best version of your lemmatizer, you should do a manual analysis of remaining errors. Describe at least 5 error types and discuss how they could be tackled in a more sophisticated lemmatizer. [ca 1/2 page]
- Lemmatizers are often implemented as finite-state-transducers (FSTs). While this kind of implementation is beyond the scope of this course, Chapter 3.5 (FSTs for Morphological Parsing) of our course book gives examples of how FSTs can be visualised. Draw an FST based on the initial lemmatizer we gave you in Lab 7 (repeated on the last page of this assignment) that can analyse the following words: `cats NOUN`, `jumped VERB`, `higher ADJ`. You can **either** draw the FST by hand, take a picture, and transform it to `.pdf`, **or** use a drawing program (e.g., xfig or MS Paint) and transform the output into `.pdf`. Regardless of how you produce the drawing, it must be included in your submission and **not** be submitted as a separate file. Please describe your drawing.
- Why is it more difficult to tag morphologically rich languages? Please reflect and motivate your answer.

[1] If you need more information about the tagset, go to http://universaldependencies.org/u/pos/index.html.

4 VG: Hidden Markov Models

Consider the following TRAIN "corpus":

```
she/PRON books/VERB trips/NOUN for/ADP kids/NOUN
she/PRON kids/VERB you/PRON all/DET the/DET time/NOUN
kids/NOUN like/VERB books/NOUN like/ADP these/PRON
```

Compute the transition and emission probabilities of a 1st order Hidden Markov Model POS-tagger using this data. Show your results in separate tables.

Now, consider the following TEST sentence:

```
kids like books
```

Report the most likely tag sequence for this sentence, as well as its probability given the TRAIN corpus. Show your work by drawing a Viterbi trellis.

5 Grading Criteria

To pass the assignment, you must meet all the basic criteria on all subparts of the assignment. To get VG, you must in addition meet some of the additional criteria for most of subparts.

Basic Criteria

- Answers are given in understandable English.
- Answers are stated clearly and coherently.
- Answers are essentially correct.

Additional Criteria

- Answers are well motivated.
- Answers are well illustrated.
- Answers reveal extensive knowledge of the textbook chapter(s).

6 Submit the assignment

Submit your assignment as a pdf file named firstname_lastname_assignment_2.pdf. It should follow the style and margins given in the example submission even if not created with LaTeX. See deadline on studentportalen.

```
import sys

def noun_lemma(word):
    if word.endswith("s"):
        return word[:-1]
    else:
        return word

def verb_lemma(word):
    if word.endswith("ed"):
        return word[:-2]
    else:
        return word

def adj_lemma(word):
    if word.endswith("er"):
        return word[:-2]
    elif word.endswith("est"):
        return word[:-3]
    else:
        return word

for line in sys.stdin:
    if line.strip():
        (word, tag) = line.strip().split("\t")
        lemma = word
        if tag == "NOUN":
            lemma = noun_lemma(word)
        elif tag == "VERB":
            lemma = verb_lemma(word)
        elif tag == "ADJ":
            lemma = adj_lemma(word)
        else:
            lemma = word
        print("{0}\t{1}\t{2}".format(word, tag, lemma))
    else:
        print()
```

Figure 1: Skeleton code for a rule-based lemmatizer, taken from Lab 7.