CE314/887 - Natural Language Engineering Assignment 1: Probabilities, Regular Expressions & Language Models

Aline Villavicencio

October 2018

Plagiarism

You are reminded that this work is for credit towards the composite mark in CE314, and that the work you submit must therefore be your own. Any material you make use of, whether it be from textbooks, the web or any other source must be ***explicitly acknowledged*** *as a comment in the program, and the extent of the reference clearly indicated.*

Part 1: Tokenization, Part-of-Speech Tagging (30%)

Q1 (20%) Create a program that reads the text from the following website and identify all the types and tokens before and after lowercasing and lemmatization. Use NLTK functions to perform these tasks, from the url reader to the tokenizer and lemmatizer.

https://www.theguardian.com/music/2018/oct/19/while-my-guitar-gently-weeps-beatles-george-harrison

The output of the program should contain the following information:

This text contains types before lemmatization:

This text contains tokens before lemmatization:

This text contains types after lemmatization:

This text contains tokens after lemmatization:

Q2 (5%) Assign part-of-speech (POS) tags to all tokens in the text used above. Use one of the implemented POS taggers in NLTK to do this.

(5%) POS taggers do not always assign correct tags to words. Identify tagging errors in the sentences above and briefly explain why these errors may have been caused.

Part 2: Regular Expressions, FSAs, and FSTs (30%)

In this part of the assignment you will do some simple information extraction, namely the identification of telephone numbers in text.

Q3 (20%) Write a regular expression that can find all telephone numbers in a text. Your expression should be able to deal with different formats, for example *+55 51 33083838*, *1206 872020*, *01206 872020* and *05679401945* as well as *+44 5679401945* and *0044 5679401945*. For full marks: include the output of a Python program that applies your regular expression to any url specified by the user, reads it and finds the telephone numbers. The output should clearly identify what the telephone number is:

```
Found a match!
Telephone: 01206872020
```

Q4 (10%) Write a FSA equivalent to the regular expression you just wrote. You can either use a drawing program or write down a transition table.

Part 3: N-gram models (40%)

General Instructions

There are two datasets given to you. Uncompress `a01_data.zip` and look at the files.

Toy dataset: The files `sampledata.txt, sampledata.vocab.txt, sampletest.txt` comprise a small toy dataset. `sampledata.txt` is the *training corpus* and contains the following:

<s> a a b b c c </s>
<s> a c b c </s>
<s> b c c a b </s>

Treat each line as a *sentence*. <s> is the start of sentence symbol and </s> is the end of sentence symbol. To keep the toy dataset simple, characters `a-z` will each be considered as a *word*. i.e. The first sentence has 8 tokens, second has 6 tokens, and the last has 7.

The file `sampledata.vocab.txt` contains the vocabulary of the training data. It lists the 3 word types for the toy dataset:

a
b
c

`sampletest.txt` is the *test corpus*.

Actual data: The files `train.txt`, `train.vocab.txt`, and `test.txt` form a larger more realistic dataset. These files have been preprocessed to remove punctuation and all words have been converted to lower case. An example sentence in the train or test file has the following form:

<s> the anglo-saxons called april oster-monath or eostur-monath </s>

Again every space-separated token is a word. The above sentence has 9 tokens. The `train.vocab.txt` contains the vocabulary (types) in the training data.

Important: Note that the <s> or </s> are not included in the vocabulary files. In this assignment, the term UNK will be used to indicate words which have not appeared in the training data. UNK is also not included in the vocabulary files but you will need to add UNK to the vocabulary while doing computations. While computing the probability of a test sentence, **any words not seen in the training data should be treated as a UNK token**.

Important: You do not need to do any further preprocessing of the data. Simply split by space you will have the tokens in each sentence.

Q5 (10%) Computing a unigram model

Use the **Toy dataset**. The vocabulary is the words in the `sampledata.vocab.txt` plus the UNK token. **Do not** include <s> and </s> in the vocabulary.

a) Compute the probabilities in a unigram language model without smoothing. Show your work for $P(a)$, $P(c)$, P(UNK). Create a table in the following format and list all of the probabilities in the unigram model.

X	**P(X)**
a	
b	
...	

b) Smooth the model using Laplace smoothing. Show your work for $P(a)$, $P(c)$, P(UNK). Show all the smoothed probabilities in a table.

Q6 (20%) Computing a bigram model

Use the **Toy dataset**. The vocabulary is the words in `sampledata.vocab.txt`, plus the UNK token and </s> symbols. <s> should be included only in the context or history. </s> should not be included in the history but only as the following word. The table below should clarify for you.

a) Compute the probabilities in a bigram language model without smoothing. Show your work for $P(b|a)$, $P(\text{UNK}|\ \text{<s>})$, $P(\text{UNK}|\text{UNK})$. Create a table in the following format and list all of the probabilities in the model.

	$P(w_i\|w_{i-1})$				
w_{i-1} \ w_i	a	b	c	UNK	</s>
a					
b					
c					
UNK					
<s>					

b) Smooth the model using Laplace smoothing. Show your work for $P(b|a)$, $P(\text{UNK}|\ \text{<s>})$, $P(\text{UNK}|\text{UNK})$. Show all the smoothed probabilities in a table.

Q7 (10%) Computing sentence probabilities

Use the **Toy dataset**. There are 5 sentences in `sampletest.txt`. Using the *smoothed* models above, compute the probability of each sentence. For unigram probability, you should ignore the <s> and </s> symbols.

a) Show your work for sentence numbers 3, 4, 5, for each model: unigram and bigram.

b) Fill in the probabilities of all the sentences in a table.

S	$P_{uni}(S)$	$P_{bi}(S)$
<s> a b c </s>		
<s> a b b c c </s>		
...		

What to submit: Report and Code

Assignments are to be made in pairs. Both students need to submit a `registration_number1_registration_number2.zip` file. The uncompressed folder should contain a report and a code directory.

Report. A file containing the answers for each of the questions.

Code. Your code should run without any arguments. It should read files in the same directory. Absolute paths must not be used. When downloaded, your code should run with a simple command such as `python LangModel.py`. A README.txt file should have a single line giving the command to run your code. Check your code by downloading your .zip file into a different machine and testing that it runs without modification.

When your code is run, it should print values in the following format.

```
---------------- Toy dataset ----------------

=== UNIGRAM MODEL ===
- Unsmoothed -
 a:0.0  b:0.0  ...

- Smoothed -
 a:0.0  b:0.0  ...
```

```
=== BIGRAM MODEL ===
- Unsmoothed -
       a    b    c  UNK  </s>
 a     0.0  ...
 b     ...
 c     ...
 UNK   ...
 <s>   ...
- Smoothed -
       a    b    c  UNK  </s>
 a     0.0  ...
 b     ...
 c     ...
 UNK   ...
 <s>   ...
== SENTENCE PROBABILITIES ==
 sent                 uprob  biprob
 <s> a b c </s>       0.0    0.0
 <s> a b b c c </s>   ...
 ...
```

Assessment criteria

What we are looking for in your answers:

Clear understanding of the concepts demonstrated by taking the right approach, giving the right formula, correct substitution and accurate answers

Ability to use concepts learned in class demonstrated by clear answers to questions which ask to analyze numbers and output

Delivering the requested software solutions demonstrated by code which satisfies the criteria, outputs the required solutions cleanly, runs without dependencies, contains proper comments. You will not be evaluated on the efficiency of your code or algorithms as long as they run in reasonable time.

The assignment, which counts for **10%** of the overall mark, can be made in pairs, and should be submitted via the electronic submission system by **week 7 (16/11), 11.59 am**. The registration numbers of both students should be clearly marked in the zip file.