NLP1 2016

Assignment 2
Due 22/11/2016
Points 30

1. A student in NLP1 would like to build a simple *spelling corrector* to solve the problem of *their* vs. *there* in English. The model would take as input sentences like **6 Points**

   He saw their cat on the street.
   He saw their was a cat on the street.

   and for each instance of *there* or *their*, predict whether the true spelling should be *there* or *their*. So, the model should predict *their* for sentence 1 above, and *there* for sentence 2 above. For the second example the model would correct the spelling mistake in the sentence.

   The student decides to use a language model for the task. Given a language model $p(w_1...w_n)$, he returns the spelling that gives the highest probability under the language model. So for example for the second sentence he implements the rule:

   If p(He saw their was a cat on the street) $>$ p(He saw there was a cat on the street)
   Then Return *their*
   Else Return *there*

   (a) The first language model the student tries is a uni-gram model. Assume that the student uses MLE (i.e. $p(w_i) = count(w_i)/N$, where $count(w_i)$ is the number of times a word occurs in a corpus and N is the total number of words in the corpus. Also assume that for every word v in the vocabulary , $count(v) > 0$). Write the general equation for sentence probability under the unigram model, i.e. $p(w_1 \dots w_n)$ Does this seem like a good solution to the *their* vs. there problem? Justify your answer.

   (b) Next, the student tries a bi-gram language model. Write the general equation for sentence probability under the bi-gram model. Why might this model be better than the model in the previous question?

2. Suppose we build a language model that makes use of the second order Markov assumption (a *trigram* language model), that is

$$P(w_1, w_2 \dots w_n) = \prod_{i=1}^{N} P(w_i | w_{i-2}, w_{i-1})$$

   Give three examples in English where English grammar indicates that this independence assumption is very clearly violated.

   3 Points

3. Suppose we want to train an HMM tagger for the task of Named Entity Recognition (NER). We are interested in only two kinds of named entities: persons (PER) and organizations (ORG), which include corporate and political entities. We have the following training data [1]

[1] Slightly modified entry for Barack Obama from Wikipedia

Barack/PER Hussein/PER Obama/PER II/PER (born August 4 , 1961) is an American politician who is the 44th and current President of the United/ORG States/ORG .

He is the first African American to hold the office and the first president born outside the continental United/ORG States/ORG .

Born in Honolulu , Hawaii , Obama/PER is a graduate of Columbia/ORG University/ORG and Harvard/ORG Law/ORG School/ORG , where he was president of the Harvard/ORG Law/ORG Review/ORG .

Obama/PER was a community organizer in Chicago before earning his law degree . He worked as a civil rights attorney and taught constitutional law at the University/ORG of Chicago/ORG Law/ORG School/ORG between 1992 and 2004 .

In this data we show only the tags for the words belonging to the person and organization categories. Assume all other words (including punctuation) have the tag OTH, which is not shown. There are 112 tokens and 69 types in the text. **10 Points**

(a) Give the transition probability matrix estimated from this training data using maximum-likelihood estimation. Don?t forget to include beginning and end of sentence markers.

(b) Now do the same but using add-one smoothing. Assume that all sentences must contain at least one word (i.e., $P(< /s > | < s >)$ is zero even in the smoothed model).

(c) Again using add-one smoothing, what are the estimates for $P(Obama|PER)$ and $P(Obama|ORG)$?

(d) Suppose we used four tags for this task: the three already mentioned, plus a LOC tag for locations. In a general text, will context always be able to disambiguate between the LOC and ORG tags?

(e) Can you think of any sources of information that might help an automatic NER system perform better, but which are *not* used by an HMM tagger? Back up your answer with examples from the text here, or give examples that could occur in another text.

4. Consider the following sentence: **3 Points**

The healthy man the lifeboats

Use a version of *bigram tagging* as described in the lectures to assign tags to this sentence, using the tags DT, N, V, ADj, based on the following frequency data. (Rows correspond to potential POS tags for the word in question; columns correspond to the POS tag of the preceding word.) Assume 'the' and 'lifeboats' can only be tagged as DT and N respectively.

healthy	DT	N	V	Adj
N	8	2	3	2
V	0	0	0	0
Adj	34	5	13	17

man	DT	N	V	Adj
N	102	45	15	86
V	0	11	4	4
Adj	0	0	0	0

5. Now use the *Viterbi* algorithm to tag the sentence **7 Points**

The healthy man the lifeboats

using the following transition and emission probabilities. That is, hand-simulate the Viterbi algorithm in order to compute the highest probability tag sequence for the given sentence. Fill in the cells in a table, where cell $[j, t]$ should contain the Viterbi value for state j at time t. Include explicit backtrace pointers in your Viterbi matrix. (Note that in the transition matrix, rows represent the previous state and columns represent the next state)

	DT	N	V	Adj
$< s >$	0.4	0.3	0.1	0.2
DT	0	0.6	0	0.4
N	0.05	0.3	0.4	0.25
V	0.4	0.3	0.1	0.2
Adj	0.1	0.5	0.2	0.2

	lifeboat	man	healthy	the
DT	0	0	0	0.5
N	0.2	0.3	0.2	0
V	0	0.1	0	0
Adj	0	0	0.4	0

Also give the joint probability of the sentence with this tag sequence. (Note that in this example, we are ignoring end of sentence markers)