Practice Exam Problems: NLP Applications

Natural Language Processing (JHU 601.465/665)

Prof. Jason Eisner

1. You are building a named-entity chunker. Here is one of your training examples, with the named entities bracketed:

   ```
   [Yoyodyne] is a company in [San Narciso, California].
   ```

   (a) [4 points] Your system will treat chunking as a tagging problem. Give the correct tagging of the above sentence under the `BIO` tagging scheme (which uses tags `B`, `I`, and `O`). (If you've forgotten the `BIO` scheme, make up something reasonable and explain it.)

   (b) [3 points] Certain tag sequences are illegal under the `BIO` tagging scheme. To take an unnecessarily long example, the tag sequence `BOIBBOO` can't correspond to any chunking. If your tagger ever returned such an illegal tag sequence, your system would not be able to extract the named entity chunks from it.

   Suppose you use an HMM tagger. How could you design it so that illegal tag sequences have probability 0? (*Hint:* Under the `BIO` tagging scheme, it is enough to prohibit certain illegal bigrams.)

2. Here is some text from today's *New York Times*. It might be hard for a computer to divide this passage into sentences. For example, the period after "Catherine T." does not mark the end of a sentence, even though it is followed by a capital letter.

> The cooperative has start-up financing from the John D. and Catherine T. MacArthur Foundation, and assistance from WTTW, a Chicago public television station. It is in talks with WBEZ, the local public radio station, about collaboration. But the first paying customer is The New York Times, which has been looking for local partners to produce journalism for expanded versions of the paper in markets around the country. Last week, The Times began a San Francisco edition, and the paper is in talks with a newly formed news cooperative in that region about handing over responsibility for producing the local pages. The Chicago News Cooperative will produce two pages, twice a week, to appear in copies of The Times that are distributed in the Chicago area. The first Chicago edition will appear on Nov. 20, The Times said.

You remember that it's sometimes wise to reduce your problem to a problem that has already been solved. A parser is already good at *dividing a sentence into phrases*, recursively. So maybe you could get a parser to also *divide a document into sentences.*

(a) [4 points] `wallstreet.gr` from homework 3 is a reasonable PCFG for parsing a *single* newspaper sentence. How would you modify `wallstreet.gr` in order to accomplish this new task?

(*Hint:* The start symbol of `wallstreet.gr` is `ROOT`, which has probability 1 of expanding via the rule `ROOT` $\rightarrow$ `S`.)

(b) [3 points] When you try your solution, the parser divides the above document into only 3 sentences. But the correct answer has 6 sentences. How can you easily adjust your solution to make the parser favor a larger number of sentences?

(c) [3 points] How can you adjust your solution to force the parser to divide the document into *exactly* 6 sentences?

(d) [5 points] Another approach, instead of using a parser, would just try to do some sort of local classification to find the sentence boundaries. Explain what you would classify and what features you would try. Consider the example paragraph at the start of the problem.

3. A certain news website allows readers to post anonymous comments on any article. The publisher hires you to write some software that will identify "good comments." The publisher plans to delete the bad comments automatically.

   (a) [3 points] The publisher insists that you must achieve 90% precision on the task of identifying good comments. What does that mean? *(circle one)*

      i. Be nice to writers. It is okay if we accidentally delete a few of the good comments, but no more than 10% of them.

      ii. Be nice to readers. It is okay if a few of the comments that appear on our website are bad, but no more than 10% of them.

      iii. Be accurate. It is okay to make some mistakes and misclassify a few comments, but no more than 10% of them.

   (b) [3 points] You tell the publisher that a good solution will need to accomplish more than just high precision. What else does it need and why?

   (c) [3 points] You decide to treat the publisher's problem as a supervised classification problem. Your first step is to collect some data. What kind of dataset do you prepare? Give details.

   (d) [6 points] In your first version, the feature vector $\vec{x}$ for a comment has one feature for each word in the vocabulary. These are count features, so x_i is the number of times that word i appears in the comment.

   Name a reasonable classification algorithm, and describe how it classifies a comment based only on this feature vector. You may use a log-linear model, a Naive Bayes model, a decision list, or any other method of your choice. But say exactly how it produces its binary classification ("good comment or bad?") in the case of this *particular* simple set of features. You may assume that the parameters of the model have been magically trained for you.

   (e) [3 points] What would your chosen method do if the entire comment consisted of OOV words, so that $\vec{x}$ is the zero vector? (This might happen if the comment is written in a foreign language.) *Note:* Don't change your method to deal with this case; just explain what the method would do if used directly.

(f) This first version of your classifier doesn't work too well, because $\vec{x}$ regards the comment as simply a "bag of words," and it is hard to determine the quality of the comment from that.

So you should add some more features (perhaps a few carefully constructed features, or perhaps a large set of similar features). In answering the questions below, make sure that you clearly define each new feature that you propose. (What is its numerical value—e.g., is it a count of something? a log-probability?)

In grading, we will give more credit to answers that we think are more likely to improve accuracy in practice. So think like a creative engineer, using any ideas you can think of from your experience of NLP or websites.

i. [5 points] Describe features that you could add in order to determine whether the comment is *well-written.* Make sure that you clearly define the numerical value of each feature.

ii. [5 points] Describe features that you could add in order to help determine whether the comment is *relevant* to the news article that it is attached to. Again, clearly define the feature values.

(g) [4 points] Your classifier still doesn't work too well, because you don't have enough supervised training data. Explain in detail how you could use *unsupervised* training data to help. (Choose a technique such as bootstrapping, EM, or Viterbi EM. Explain how you would apply it, and on what data.)

4. [30 total points] A recent bestseller (2009) was called *Going Rogue: An American Life*, by Sarah Palin. However, it has been widely reported that the text was mainly crafted by a professional writer, Lynn Vincent. (It is common for politicians who write books to have help from "ghostwriters.")

   You would like to use NLP to investigate these reports. You have

   - a large corpus V of text definitely written by Vincent (from Vincent's published books)
   - a small corpus P of text definitely written by Palin (from a small corpus of Palin's email[1])

   Assume for the moment that the book B was *entirely* written by either Palin or Vincent. You would like to determine which one it was.

   You plan to issue a press release if your experiments "prove" that Palin wrote the book herself, contrary to reports. Specifically, you will issue a press release if your posterior belief rises to

   $$p(\text{Author} = \text{Palin} \mid \text{Text} = B) > 0.9$$

   (a) [2 points] As your first attempt, you use Bayes' Theorem to obtain a formula for your posterior belief. What is that formula?

   $p(\text{Author} = \text{Palin} \mid \text{Text} = B) =$ ______________________________

   (b) [3 points] To help you evaluate that formula, you train trigram models p_V and p_P on your two corpora V and P. Under what conditions will you then issue a press release?

   Your answer should be written as an inequality that uses $p_V(B)$ and $p_P(B)$ together with your prior belief that Palin wrote the book. Your prior (based on previous news reports) is

   $$p_{\text{prior}}(\text{Author} = \text{Palin}) = 0.1$$

   and as mentioned above, your threshold for a press release is 0.9.

   (c) [3 points] As your second attempt, you switch from this generative model to a discriminative model. In other words, you skip Bayes' Theorem and switch directly to modeling $p(\text{Author} = \text{Palin} \mid \text{Text} = B)$. You model this and similar conditional probabilities with a log-linear model of the form

   $$p(\text{Author} = A \mid \text{Text} = T) = \frac{1}{Z} \exp \sum_i f_i(A, T) \cdot \theta_i$$

[1] In September 2008, hackers guessed the password reminder prompts to Palin's accounts `gov.palin@yahoo.com` and `gov.sarah@yahoo.com`, and released screenshots showing emails from her.

Your feature functions f_i include only unigram, bigram, and trigram counts (even though an advantage of this framework is that you could include other features as well).

The normalizing constant Z depends on T and $\vec{\theta}$. Give a formula for Z.

(d) When training the parameters of the two generative models, you are essentially trying to maximize[2]

$$\prod_{s \in V} p(\text{Text} = s \mid \text{Author=Vincent}) \cdot \prod_{s \in P} p(\text{Text} = s \mid \text{Author=Palin})$$

where s ranges over the sentences in a corpus. By contrast, when training the parameters $\vec{\theta}$ of the discriminative model, you are trying to maximize

$$\prod_{s \in V} p(\text{Author=Vincent} \mid s) \cdot \prod_{s \in P} p(\text{Author=Palin} \mid s)$$

(In both cases, you typically modify this maximization problem to do some smoothing or regularization of the parameters, but never mind that.)

i. [2 points] Palin does *not really* generate her text from an n-gram model (whatever bloggers may say about her). Neither does Vincent.
Is that an argument to choose generative training, to choose discriminative training, or does it not clearly indicate either choice? Explain.

ii. [2 points] Notice that V is much bigger than P.
Is that an argument to choose generative training, to choose discriminative training, or does it not clearly indicate either choice? Explain.

iii. [5 points] Suppose you have a third corpus that contains a mixture of Vincent sentences and Palin sentences. These sentences are *not* labeled with their authors, so you would like to use EM on this "raw" dataset to improve your model.
Explain how you would do this, including your choice of either the generative or the discriminative model.

[2] This is achieved in practice by training the p_V model's parameters to maximize the first factor and the p_P model's parameters to maximize the second factor. But that serves to maximize their product.

iv. [3 points] This is really a text classification problem, and there are are other machine learning options for text classification.
Would a decision list be trained more like the generative model or the discriminative model above?

How about a decision tree?

How about a clustering-based method?

(e) It is silly to think that either Palin wrote *all* the text or Vincent did. Describe some reasonable generative model that would explain how they *collaborated* on the book.

(Some options would include a mixture model where each one wrote *some* of the sentences; a PCFG where each nonterminal has a feature "author=Palin" or "author=Vincent"; or a noisy channel where one of them wrote the book and the other edited it.)

i. [7 points] Describe how you would compute $p(\text{Text} = B)$ under your model (assuming you had learned the parameters). Remember that your model is based on a random process for how Palin and Vincent would write together. Thus, $p(\text{Text} = B)$ is the probability that they would have generated this book *somehow*; it sums over hidden variables that model authorship, syntactic structure, etc.

ii. [3 points] How would you use your model to quantify what fraction of the book was written by Palin?