Practice Exam Problems:
Probability Models

Natural Language Processing (JHU 601.465/665)

Prof. Jason Eisner

1. The Tablish language uses the sixteen letters `a`, `b`, `c`, ... `p`, and no other symbols. You build a unigram letter model of Tablish in which each of these letters has probability $\frac{1}{16}$.

   (a) What single word describes this probability distribution? Uniform Distribution

   (b) What is the cross-entropy per letter of a Tablish text under your model? __________

   (c) What is the perplexity per letter of a Tablish text under your model? __________

   (d) What is the cross-entropy per letter of "`an English text`" under your model? __________

   (e) What is the perplexity per letter of "`an English text`" under your model? __________

2. True or false? On a test corpus where the average word has 6 letters, a model's perplexity per word must be exactly 6 times its perplexity per letter. (To avoid worrying about whether spaces count as letters, let's assume the corpus is in Chinese, which has no spaces.)

   (a) true

   (b) false

3. I am the proud owner of a weighted coin that comes up heads more often than tails.

   (a) The following columns represent actual sequences of flips that I obtained. For each sequence, fill in the two estimates of $p(\texttt{H})$.

Observed sequence	HHHH	HHHH	THHH	HTHH	THHT	HTHT
unsmoothed estimate of $p(\texttt{H})$						
add-1-smoothed estimate of $p(\texttt{H})$						

   (b) In practice, you would observe *only one* of these sequences of flips. We showed 6 actual sequences to explore some of the different scenarios for what you might observe ("alternate universes"). Averaging over these 6 scenarios, what would be your *average* unsmoothed estimate of $p(\texttt{H})$? __________

   How about your *average* smoothed estimate? __________

   (c)
   - Which technique—smoothed or unsmoothed—has less bias? __________
   - Which technique—smoothed or unsmoothed—has less variance? __________
   - Which *average* from part (b) is likely to be closer to the truth? __________
   - Why might someone prefer the other estimation technique anyway? (Answer in plain language, not statistical jargon.)

   (d) Suppose you estimate (somehow, from some training data) that $p(\texttt{H}) = 0.8$, $p(\texttt{T}) = 0.2$. You evaluate this unigram model by obtaining a 6-flip test sequence `HHHHTT`.

   What is the cross-entropy per character of your estimated model on this test set? (Answer in bits. Simplify your answer as much as is reasonable without a calculator.)

   (e) Your 6-flip test sequence `HHHHTT` might not be typical. After all, if you had had a different test sequence, the cross-entropy number would have come out differently (that is, there is some variance in your method of estimating cross-entropy). So maybe you can't trust the cross-entropy number that you got . . .

   What is the real cause of this worry? What would be an easy way to get a more accurate and replicable estimate of the cross-entropy of your model with respect to the true (but unknown) behavior of this coin?

4. In a bigram language model, each word depends on the *single previous word*. Thus, the probability of `Sing a song of sixpence` is modeled as

$$p(\vec{w}) = p(w_0) \cdot p(w_1 \mid w_0) \cdot p(w_2 \mid w_1) \cdot p(w_3 \mid w_2) \cdot p(w_4 \mid w_3) \cdot p(w_5 \mid w_4) \cdot p(w_6 \mid w_5)$$

where $\vec{w} = (w_0, w_1, w_2, w_3, w_4, w_5, w_6) = (\text{BOS}, \texttt{Sing}, \texttt{a}, \texttt{song}, \texttt{of}, \texttt{sixpence}, \text{EOS})$.[1]

In class, we justified this formula using the chain rule of conditional probability together with certain choices about how to back off (i.e., conditional independence assumptions). Which of the following alternative models could be justified using the same technique?

(*Note:* I'm not asking which of these are good models of natural language; that depends on how accurate the conditional independence assumptions are. I am only asking which ones can be formally justified.)

(circle all that apply)

(a) Each word depends only on the *single following* word:

$$p(w_0 \mid w_1) \cdot p(w_1 \mid w_2) \cdot p(w_2 \mid w_3) \cdot p(w_3 \mid w_4) \cdot p(w_4 \mid w_5) \cdot p(w_5 \mid w_6) \cdot p(w_6)$$

(b) Each word depends only on the *previous word* and the *next word*:

$$p(w_0 \mid w_1) \cdot p(w_1 \mid w_0, w_2) \cdot p(w_2 \mid w_1, w_3) \cdot p(w_3 \mid w_2, w_4) \cdot p(w_4 \mid w_3, w_5) \\ \cdot p(w_5 \mid w_4, w_6) \cdot p(w_6 \mid w_5)$$

(c) Each word depends only on the *previous uncommon word*, meaning the previous word that does not appear on a given list (e.g., the 100 most frequent unigrams of English). The uncommon words in `Sing a song of sixpence` are w_1, w_3, and w_5 as well as w_0 and w_6. As a result, the formula below uses $p(w_3 \mid w_1)$, with the intent that `song` should depend on `sing`, rather than on `a`.

$$p(w_0) \cdot p(w_1 \mid w_0) \cdot p(w_2 \mid w_1) \cdot p(w_3 \mid w_1) \cdot p(w_4 \mid w_3) \cdot p(w_5 \mid w_3) \cdot p(w_6 \mid w_5)$$

(d) Each word depends only on *the number of uncommon words that precede it* (which reflects its position in the sentence).

$$p(w_0) \cdot p(w_1 \mid 0) \cdot p(w_2 \mid 1) \cdot p(w_3 \mid 1) \cdot p(w_4 \mid 2) \cdot p(w_5 \mid 2) \cdot p(w_6 \mid 3)$$

(e) none of the above

[1] As usual, BOS and EOS are special symbols denoting "beginning of sentence" and "end of sentence."

5. This question asks you to build and use a probability model in a real-world situation. You should be thinking about concepts like Bayes' Theorem, noisy channel models, modeling a joint probability by using the chain rule and then backing off via conditional independence assumptions, etc.

   You would like to build a system that catches plagiarism in humanities courses. When a student submits a written essay, you will check each sentence $\vec{W} = w_1 w_2 \ldots w_n$ to see whether it was plagiarized.

   In the old days, this was easy. You had a large corpus $\mathcal{C}$ of sentences from existing essays. If $\vec{W}$ appeared in $\mathcal{C}$, then you classified the sentence as plagiarized.

   But now students are trickier. A cheating student might first copy a sentence from, say, my own paragraph above:

   - $\vec{V}$ = `In the old days, this was easy.`

   To disguise her theft, however, she randomly right-clicks on the words `old`, `this`, and `easy` to bring up a list of synonyms in her word processor:[2]

   - syn(`old`) = {`old`, `aged`, `elderly`, `senior`, `aging`}
   - syn(`days`) = {`days`, `daylights`, `daytimes`, `lights`, `sunlights`, `sunshines`}
   - syn(`was`) = {`was`}

   She replaces each word v_i that she clicked on with a synonym chosen randomly from syn(v_i). It is the resulting sentence that she includes in her essay:

   - $\vec{W}$ = `In the senior sunshines, this was easy.` *(note that $\vec{W} \notin \mathcal{C}$)*

   Now, the student thinks she has won, because we won't be able to find $\vec{W}$ in $\mathcal{C}$. But $\vec{W}$ does seem awfully strange! It doesn't look like normal English.[3] So maybe we can still detect that it is plagiarized.

   We can treat this as a language ID problem—is the sentence $\vec{W}$ written in English or "Synglish"? (In this problem, we use "Synglish" to denote the variant of English produced by the synonym-substitution procedure above.)

[2] This trick is becoming more common, according to popular press articles in August 2014. http://slate.me/1uxbPi3 appeared in *Slate* (an American online magazine), and drew on a previous article from *The Guardian* (a British newspaper). Truth or urban legend?

[3] This student will certainly get an "F" on her essay in any case, because her sentences will make no sense. But it's still important to distinguish plagiarism from mental illness, since the remedies are different.

(a) Your first thought is to train a log-linear model of the conditional probability distribution $p(y \mid x)$ where $y \in \{\texttt{English}, \texttt{Synglish}\}$ and $x = \vec{W}$. Why wouldn't this be straightforward? Because, the features required for log-linear models, there is no particular way of creating features to distinguish between languages if both languages are same. As the Stylish is just English synonyms, it's harder to differentiate. Also, we don't have enough Singlish data to train.

(b) So in the rest of this question, we'll try to use models that incorporate our understanding of how the observed sentence $\vec{W}$ got generated. (The previous question did not do this; it just took $\vec{W}$ as given, and conditioned on it.)

Assume that you have good language models for both languages. So given a sentence $\vec{W}$, you can compute both $p_{\text{English}}(\vec{W})$ and $p_{\text{Synglish}}(\vec{W})$.

This should help you to compute conditional probabilities such as $p(\text{Synglish} \mid \vec{W})$. Thus, please give a formula for the following "odds ratio":

$$\frac{p(\text{Synglish} \mid \vec{W})}{p(\text{English} \mid \vec{W})} = \frac{\text{p(W|Synglish) * p(Synglish)}}{\text{p(W|English) * p(English)}} = \frac{\text{p\_Synglish(W) * p(Synglish)}}{\text{p\_English(W) * p(English)}}$$

For example, if the odds ratio is 2, then $\vec{W}$ is twice as likely to be Synglish as English (so the probabilities of Synglish and English are 2/3 and 1/3).

(c) But where do we get our Synglish language model? Define a formula for $p_{\text{Synglish}}(\vec{W})$. I've already sketched the process the student uses to generate a Synglish sentence. (What dice does she have to roll to get $\vec{W}$?) So all you have to do is to write this down as a probability formula.

Actually, my description above is not absolutely precise—it glosses over details such as the weights of the student's dice. Your formula will have to be completely precise. So you should just fill in those details in a reasonable way. Nothing fancy is expected.

$$p_{\text{Synglish}}(\vec{W}) = \text{SUM\_v p(W,V) = SUM\_v p(W\_1|V\_1) * p(W\_1|V\_1) * p(W\_1|V\_1) * ....}$$

Model (A) -> 100-gram model
Model (B) -> 3-gram model

(d) In the process sketched above, the student starts by choosing an English sentence $\vec{V}$. There are at least two reasonable ways to model this. You could say either (A) that $\vec{V}$ is a random sentence from our large essay corpus $\mathcal{C}$, or (B) that $\vec{V}$ is a random sentence from a *trigram model* that has been trained on $\mathcal{C}$.[4]

i. Model $\underbrace{(A) \quad (B)}_{\text{circle one}}$ suffers from greater <u>bias</u> because ... ?

Model (A) has more Bias

ii. Model $\underbrace{(A) \quad (B)}_{\text{circle one}}$ suffers from greater <u>variance</u> because ... ?

Model (B) Has more variance

iii. How could you compromise between these ("get the best of both")?

Model (A) with backoff and smoothing.

(e) You've caught a plagiarist! Your model is sure that $\vec{W}$ is Synglish, not English. Unfortunately, the accused student has hired a fancy lawyer.

Fortunately, you are a fancy computer scientist, so you can fight back. :-) To convince a jury, you need to show them your theory of the crime—the original English sentence $\vec{V}$ that was most likely to have given rise to $\vec{W}$.

i. You are looking for the $\vec{V}$ that maximizes what quantity?
(Your answer should be a short formula. You don't have to repeat material from your answer to question 5c.) p(V|W) = p(V) * p(W|V)

ii. What is the standard name for this kind of maximization problem?

Decoding

[4] To be clear, we don't really believe that the *student* is using randomness to select the sentence. Surely she has some reason for picking the $\vec{V}$ that she does—e.g., perhaps it is relevant to her assigned essay topic. However, we don't know what her reasons are. So the randomness represents *our* uncertainty about what she will do, while still modeling our belief that she's more likely to pick some sentences as $\vec{V}$ than others.