Questions for Flipped Classroom Session of COMS 4705 Week 6, Spring 2018. (Michael Collins)

Question 1 In this question we consider the problem of mapping a sentence to an underlying sequence of tags, using a log-linear tagger. The input to the tagger is a sequence of words $x_1 x_2 \dots x_N$, where each x_i is an English word. The output from the tagger is a sequence of tags $y_1 y_2 \dots y_N$. Each tag y_i can take any one of four possible states, A, B, C or D. Looking at the data, we notice that tag sequences follow the following rules:

- The tag y_1 is always equal to either A or B.
- For all tag bigrams y_j, y_{j+1}, we either have $y_j \in \{A, B\}$, and $y_{j+1} \in \{C, D\}$; or we have $y_j \in \{C, D\}$, and $y_{j+1} \in \{A, B\}$. (That is, we never see the tag bigrams AA, AB, BA, BB, CC, CD, DC or DD.)
- If the j'th word x_j in the sequence has an odd number of letters, its tag y_j is always equal to either A or C. If the j'th word has an even number of letters, its tag is always equal to either B or D.

We will use a log-linear bigram tagger to map sentences to tag sequences. The model takes the form

$$p(y_1 \dots y_N | x_1 \dots x_N) = \prod_{j=1}^{N} p(y_j | y_{j-1}, x_1 \dots x_N, j)$$

where

$$p(y_j | y_{j-1}, x_1 \dots x_N, j) = \frac{\exp\{v \cdot f(y_{j-1}, x_1 \dots x_N, j, y_j)}{\sum_{y \in \{A,B,C,D\}} \exp\{v \cdot f(y_{j-1}, x_1 \dots x_N, j, y)\}}$$

and $f(y_{j-1}, x_1 \dots x_N, j, y) \in \mathbb{R}^d$ is a feature vector, and $v \in \mathbb{R}^d$ is a parameter vector, where d is the number of parameters.

Question: Give a feature-vector definition $f(y_{j-1}, x_1 \dots x_N, j, y) \in \mathbb{R}^d$ that allows the model to perfectly model the constraints given above.

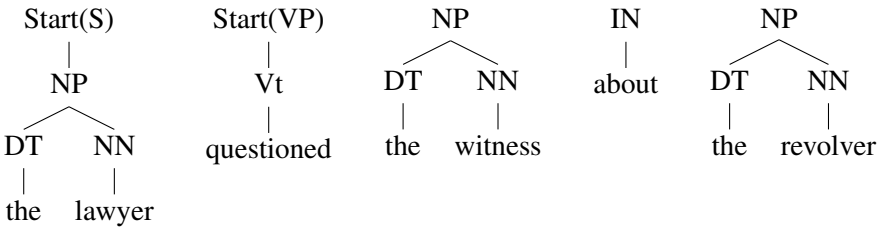
Question 2 Recall that in Ratnaparkhi's history-based model for parsing, the parser alternates Between two classes of actions:

- Join(X) or Start(X), where X is a label (NP, S, VP etc.)
- Check=YES or Check=NO

The meaning of these actions is as follows:

- Start(X) starts a new constituent with label X
  (always acts on leftmost constituent with no start or join label above it)
- Join(X) continues a constituent with label X
  (always acts on leftmost constituent with no start or join label above it)
- Check=NO does nothing
- Check=YES takes previous Join or Start action, and converts it into a completed constituent

Now assume that we are parsing the sentence *the lawyer questioned the witness about the revolver*, and we have reached the following parse state:



Question 2(a): What is the next action if the complete parse has the correct PP attachment with *about the revolver* modifying the verb *questioned*? What is the next action for the incorrect parse with *about the revolver* modifying *the witness*?

Question 2(b): How would you define the feature-vector definition that maps a parse state, and an action to a feature vector, in such a way that PP attachment decisions are sensitive to the verb, noun, preposition, and second noun, in cases such as the example above (i.e., the model should base the attachment decision on *questioned, witness, about, revolver* in the above example).

Question 3. Consider a log-linear model, with a feature-vector definition $f(x, y) \in \mathbb{R}^d$, a parameter vector $v \in \mathbb{R}^d$, a set of possible labels $\mathcal{Y}$, and the model then defined as

$$p(y|x; v) = \frac{\exp\{v \cdot f(x, y)\}}{\sum_{y' \in \mathcal{Y}} \exp\{v \cdot f(x, y')\}}$$

Assume we have a set of training examples $(x^{(i)}, y^{(i)})$ for $i = 1 \ldots n$. Define the following functions of the parameters:

$$L_1(v) = \sum_{i=1}^{n} \log p(y^i|x^i; v)$$

$$L_2(v) = \sum_{i=1}^{n} \log p(y^i|x^i; v) - \frac{\lambda}{2} \sum_{j=1}^{d} (v_j)^2$$

$$L_3(v) = \sum_{i=1}^{n} \log p(y^i|x^i; v) - \lambda \times \sum_{j=1}^{d} |v_j|$$

where $|v_j|$ is the absolute value of v_j.

Answer the following true/false questions below. For each question make sure to give a justification for your answer: you will gain at most half marks for a correct answer without a correct justification.

Question 3a: True or False: For any parameter v_j, for any value of the parameters $v_1 \ldots v_d$,

$$\frac{\partial L_2(v)}{\partial v_j} > \frac{\partial L_1(v)}{\partial v_j}$$

Question 3b: True or False: For any parameter v_j such that $v_j \neq 0$,

$$\frac{\partial L_3(v)}{\partial v_j} = \frac{\partial L_1(v)}{\partial v_j} - \lambda \times \text{sign}(v_j)$$

where $\text{sign}(v_j)$ is 1 if $v_j > 0$, and $\text{sign}(v_j)$ is -1 if $v_j < 0$.

Question 3c: True or False: Define

$$v^1 = \arg\max_v L_1(v)$$

and

$$v^2 = \arg\max_v L_2(v)$$

(here we assume that both v^1 and v^2 exist).

True or False?: for any training set,

$$L_1(v^1) \geq L_2(v^2)$$