Midterm Exam for Natural Language Processing
March 9, 2017

Name: ____________________________________________
Net ID __________________

Instructions
There are 7 questions, each will be worth 15 points for a total of 105 points. You will have approximately
1:15 minutes to complete this test.
The test materials will include this printout and, optionally, one blank blue-covered booklet. I suggest that
you fill in all answers directly on this printout. The blue booklet is available if you would like scrap paper, or
if you run out of space on the test and need somewhere else to write your answers. You also have the option
of putting all your answers in the blue booklet, if you prefer. Whatever you do, please make it clear. Write me
text notes about where to find particular questions if necessary. As long as I can find and understand your answers,
it’s fine. If you use the blue booklet, please write your name on this as well, so I can identify it as yours if it
gets separated from this printout.
This test is an open book/open notes test: Please feel free to bring your text book, your notes, copies of class
lectures and other reading material to the test. A calculator is also permitted and it is OK to look at materials
on the web in order to read helpful information, being mindful of the time limit. Just don’t use a program that
solves a problem for you, e.g., do not find a part of speech tagger and run it if asked to manually annotate
mark parts of speech – that WOULD be cheating.

Answer all questions on the test. If you show your work and you make a simple arithmetic mistake, but
it is clear you knew how to do it, you will get partial credit.

Question 1. Write a regular expression to identify names of kings, queens and other names of people holding
royal titles. Your expression should match person names accompanied by an adjacent royal title—the title
should either: (a) immediately precede the name; or (b) immediately follow the name and a comma. The
expression should match all the examples below and generalize to cover some other royal names. It should
also be specific enough that it does not match most non-royal names. In particular, it should not match any
part of the strings: a) My dog Prince, b) Disney Princess bedroom set, or c) Queens, Prince, and Kings
1. King David
2. Princess Mononoke
3. Queen Amidala
4. King Yoganarendra Malla
5. Emperor Sun Zhi
6. Empress Dowager Cixi
7. King George III
8. Duke Boleslaw Krzywousty
9. Diana, Princess of Wales
10. Henry Plantagenet, Duke of Normandy
11. Seth, Emperor of Azania
12. Prince Edward, Duke of Kent
13. George I, King of the Hellenes
**Table 1: Penn Treebank POS tags**

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>Coordinating conjunction</td>
<td>RB</td>
<td>Adverb</td>
</tr>
<tr>
<td>CD</td>
<td>Cardinal number</td>
<td>RBR</td>
<td>Adverb, comparative</td>
</tr>
<tr>
<td>DT</td>
<td>Determiner</td>
<td>RBS</td>
<td>Adverb, superlative</td>
</tr>
<tr>
<td>EX</td>
<td>Existential there</td>
<td>RP</td>
<td>Particle</td>
</tr>
<tr>
<td>FW</td>
<td>Foreign word</td>
<td>SYM</td>
<td>Symbol</td>
</tr>
<tr>
<td>IN</td>
<td>Preposition or subordinating conjunction</td>
<td>TO</td>
<td>to</td>
</tr>
<tr>
<td>JJ</td>
<td>Adjective</td>
<td>UH</td>
<td>Interjection</td>
</tr>
<tr>
<td>JJR</td>
<td>Adjective, comparative</td>
<td>VB</td>
<td>Verb, base form</td>
</tr>
<tr>
<td>JJS</td>
<td>Adjective, superlative</td>
<td>VBD</td>
<td>Verb, past tense</td>
</tr>
<tr>
<td>LS</td>
<td>List item marker</td>
<td>VBG</td>
<td>Verb, gerund or present participle</td>
</tr>
<tr>
<td>MD</td>
<td>Modal</td>
<td>VBN</td>
<td>Verb, past participle</td>
</tr>
<tr>
<td>NN</td>
<td>Noun, singular or mass</td>
<td>VBP</td>
<td>Verb, non-3rd person singular present</td>
</tr>
<tr>
<td>NNS</td>
<td>Noun, plural</td>
<td>VBZ</td>
<td>Verb, 3rd person singular present</td>
</tr>
<tr>
<td>NNP</td>
<td>Proper noun, singular</td>
<td>WDT</td>
<td>Wh-determiner</td>
</tr>
<tr>
<td>NNPS</td>
<td>Proper noun, plural</td>
<td>WP</td>
<td>Wh-pronoun</td>
</tr>
<tr>
<td>PDT</td>
<td>Predeterminer</td>
<td>WPS</td>
<td>Possessive wh-pronoun</td>
</tr>
<tr>
<td>POS</td>
<td>Possessive ending</td>
<td>WRB</td>
<td>Wh-adverb</td>
</tr>
<tr>
<td>PRP</td>
<td>Personal pronoun</td>
<td>PU</td>
<td>Punctuation</td>
</tr>
<tr>
<td>PRPS</td>
<td>Possessive pronoun</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Question 2.** Manually process the following sentence in two ways, filling in the columns in the chart below:

> It was a shy nocturnal creature with the appearance of a medium-size dog, except for its abdominal pouch and dark stripes.¹

I have tokenized this sentence and placed the tokens in the first column in the table below. Fill in the second column with PENN TREEBANK parts of speech (POS) tags, as per Table 1 (unlike conventional Penn Treebank Tags, all punctuation is marked *PU*). In the third column, enter a BIO tag indicating whether a token is beginning a noun group (B), inside a noun group (I) or outside of a noun group (O). Remember not to include right modifiers as noun groups are not full NPs. If you are uncertain about any part of speech assignment, include a short note why you chose the tag you did.

¹Adapted from Wikipedia article about creatures known as a thylacines, aka, “Tasmanian Tigers.”
It was a shy nocturnal creature with the appearance of a medium size dog except for its abdominal pouch and dark stripes.
**Question 3.** Draw a Phrase Structure Tree analyzing the same sentence you analyzed in Question 2, i.e.,

It was a shy nocturnal creature with the appearance of a medium-size dog, except for its abdominal pouch.

You should assume the same POS tags you used in Question 2. Assume that the words *except* and *for* together form a unit that “acts” like a single preposition. Note that the sentence has been shortened slightly (leaving out the words “and dark stripes”)**
Question 4: Fill in the CKY chart below for sentence

Hope springs eternal

assuming the rules below. Remember that the rows of the chart represent start positions and the columns represent end positions.

1. $S \rightarrow NP \ VP$
2. $NP \rightarrow NN$
3. $NP \rightarrow NNS$
4. $NP \rightarrow NNP$
5. $NP \rightarrow NN \ NN$
6. $NP \rightarrow NN \ NNS$
7. $VP \rightarrow VB$
8. $VP \rightarrow VBZ$
9. $VP \rightarrow VBZ \ JJ$
10. $NN \rightarrow hope$
11. $NNP \rightarrow hope$
12. $VB \rightarrow hope$
13. $NNS \rightarrow springs$
14. $VBZ \rightarrow springs$
15. $JJ \rightarrow eternal$

<table>
<thead>
<tr>
<th></th>
<th>hope</th>
<th>springs</th>
<th>eternal</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

hope springs eternal
**Question 5.** For this question, compute the probability of the sentence

*There was an old person from Rome*

assuming the following language model, using frequencies of words in a copy of *Edward Lear’s Book of Nonsense* available through Project Gutenberg:

- The probability of a sentence is the product of the probability of all the tokens in that sentence and the probability that the sentence ends (probability of an end_sentence token). Thus if N is the length of the sentence, multiply N+1 factors for the total probability.

- The probability of each token is computed based on the bigram, unigram and out of vocabulary frequencies found in the training corpus.

- The following backoff model is assumed:
  - Use the bigram probability of each token, given the previous token, if available. If the previous token is OOV, take the bigram of the current token given that the previous token is OOV.
  - Otherwise, use the unigram probability of the token, if available.
  - Otherwise, use the unigram OOV probability.

Use the following information to calculate these probabilities:

- Bigram frequencies for bigrams ending with tokens in the sentence. Each bullet lists bigrams beginning with the same first token. B_Sentence represents the beginning of the sentence and *oov* represents out of vocabulary words.\(^2\)

  - \(B_{\text{Sentence}} + *\text{oov}* \rightarrow 124; B_{\text{sentence}} + \text{and} \rightarrow 35; B_{\text{sentence}} + \text{but} \rightarrow 35; \)
    
  - \(B_{\text{sentence}} + \text{he} \rightarrow 37; B_{\text{sentence}} + \text{she} \rightarrow 20; B_{\text{sentence}} + \text{so} \rightarrow 19; B_{\text{sentence}} + \text{that} \rightarrow 46; \)
    
  - \(B_{\text{sentence}} + \text{there} \rightarrow 113; B_{\text{sentence}} + \text{till} \rightarrow 11; B_{\text{sentence}} + \text{to} \rightarrow 11; \)
    
  - \(B_{\text{sentence}} + \text{when} \rightarrow 26; B_{\text{sentence}} + \text{which} \rightarrow 25; B_{\text{sentence}} + \text{who} \rightarrow 65; \)
    
  - \(B_{\text{sentence}} + \text{whose} \rightarrow 30; \)

  - \(\text{there} + \text{was} \rightarrow 113; \)

  - \(\text{are} + *\text{oov}* \rightarrow 3; \text{are} + \text{of} \rightarrow 1; \text{are} + \text{you} \rightarrow 2; \)

  - \(\text{old} + *\text{oov}* \rightarrow 3; \text{old} + \text{derry} \rightarrow 1; \text{old} + \text{lady} \rightarrow 4; \text{old} + \text{man} \rightarrow 91; \text{old} + \text{person} \rightarrow 42; \)

  - \(\text{person} + \text{of} \rightarrow 51; \text{person} + \text{whose} \rightarrow 1; \)

  - \(\text{from} + *\text{oov}* \rightarrow 2; \text{from} + \text{his} \rightarrow 1; \text{from} + \text{the} \rightarrow 2; \text{from} + \text{this} \rightarrow 1; \text{from} + \text{turkey} \rightarrow 1; \)

  - \(*\text{oov}* + ! \rightarrow 13; *\text{oov}* + *\text{oov}* \rightarrow 67; *\text{oov}* + , \rightarrow 147; *\text{oov}* + . \rightarrow 117; *\text{oov}* + ; \rightarrow 69; *\text{oov}* + ? \rightarrow 5; *\text{oov}* + \text{a} \rightarrow 7; *\text{oov}* + \text{all} \rightarrow 5; *\text{oov}* + \text{and} \rightarrow 15; *\text{oov}* + \text{by} \rightarrow 9; *\text{oov}* + \text{E}_{\text{sentence}} \rightarrow 5; *\text{oov}* + \text{from} \rightarrow 5; *\text{oov}* + \text{her} \rightarrow 5; *\text{oov}* + \text{his} \rightarrow 12; *\text{oov}* + \text{in} \rightarrow 6; *\text{oov}* + \text{of} \rightarrow 26; *\text{oov}* + \text{old} \rightarrow 24; *\text{oov}* + \text{that} \rightarrow 17; *\text{oov}* + \text{the} \rightarrow 9; *\text{oov}* + \text{to} \rightarrow 17; *\text{oov}* + \text{was} \rightarrow 9; *\text{oov}* + \text{with} \rightarrow 9; *\text{oov}* + \text{young} \rightarrow 6; \)

- The unigram frequencies of each token, including *oov* and E_Sentence (end of sentence). Unigram frequencies are: \(B_{\text{sentence}} \rightarrow 688; \text{there} \rightarrow 113; \text{are} \rightarrow 6; \text{old} \rightarrow 141; \text{person} \rightarrow 52; \text{from} \rightarrow 7; *\text{oov}* \rightarrow 701; \text{E}_{\text{sentence}} \rightarrow 688 \)

- **There are a total of 5884 words (tokens) in the corpus.**

\(^2\)For bigrams such that the first item is B_Sentence, only bigrams with frequency greater than 10 are included. For bigrams where the first item is *oov*, only bigrams with frequency greater than 5 are included.
**Question 6.** Using the Viterbi algorithm with the transition and likelihood probabilities below: (a) calculate the probability that the sequence of words *free ducks* will be assigned the parts of speech: VB NNS (as in the command telling someone to let some ducks free); (b) calculate the probability that the sequence will be assigned the parts of speech: JJ NNS, as in the noun phrase that refers to ducks that do not cost any money. **Hint:** You do not have to fill out the entire table to calculate these two probabilities.

<table>
<thead>
<tr>
<th>Likelihood Probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>POS</strong></td>
</tr>
<tr>
<td>NNS</td>
</tr>
<tr>
<td>VB</td>
</tr>
<tr>
<td>VBP</td>
</tr>
<tr>
<td>VBZ</td>
</tr>
<tr>
<td>JJ</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transition Probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>VB</strong></td>
</tr>
<tr>
<td>.25</td>
</tr>
<tr>
<td>.25</td>
</tr>
<tr>
<td>.25</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Viterbi Table</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>START</strong></td>
</tr>
<tr>
<td>START</td>
</tr>
<tr>
<td>NNS</td>
</tr>
<tr>
<td>VB</td>
</tr>
<tr>
<td>VBP</td>
</tr>
<tr>
<td>VBZ</td>
</tr>
<tr>
<td>JJ</td>
</tr>
<tr>
<td>END</td>
</tr>
</tbody>
</table>
Question 7. Computer precision, recall and F-measure given the following answer and system output:

Answer Key

1. King David
2. Princess Mononoke
3. Queen Amidala
4. King Yoganarendra Malla
5. Emperor Sun Zhi
6. Empress Dowager Cixi
7. King George III
8. Duke Boleslaw Krzywousty
9. Diana, Princess of Wales
10. Henry Plantagenet, Duke of Normandy
11. Seth, Emperor of Azania
12. Prince Edward, Duke of Kent
13. George I, King of the Hellenes

System Output

1. Princess Mononoke
2. King Kong
3. Queen Amidala
4. Disney Princess Bedroom Set
5. Emperor Sun Zhi
6. Empress Dowager Cixi
7. Duke Boleslaw Krzywousty
8. Diana, Princess of Wales
9. Prince Edward, Duke of Kent
10. King Kullen Supermarket