CS114: Quiz 1 Review

March6, 2012

James Pustejovsky

Brandeis University

Additional slides courtesy of Jurafsky & Martin, Marie Meteer and , Ray Mooney

Question 1

1. Ambiguity is the bane of Computational Linguistics. Give one example of each of the following types of ambiguity: (10 pts. 90%)

- Phonological
  - *There their*
- Syntactic
  - *He saw a boy with a telescope*
- Semantic
  - *Time flies*

Question 2

2.a. Characterize the difference between derivational morphology and inflectional morphology.

- *Derivational morphology usually changes the part of speech and affects the meaning of the word (e.g. known vs. unknown, compute vs. computer vs. computational).*
- *Inflectional morphology indicates grammatical information such as number and tense.*

Question 2

2c. How could a FSA of derivational morphology be used in a spell checker? Why would you want to use it?

- *Using an FSA rather than listing all the words would save space:*
- *Compute +er +ation +ationally*

2d. Someone told you that you could use inflectional morphology to differentiate nouns from verbs in English. Do you believe them? Why or why not?

- *No. The same affixes (e.g. "s") are used for both nouns and verbs.*
  - *Is "dogs" a noun or a verb?*

Question 3

2. a. Find the minimum edit distance between "door" and "for".

R	3	4	3	4	3	
O	2	3	2	3	4	
F	1	2	3	4	5	
	0	1	2	3	4	
		D	O	O	R	

Question 3

3.b Give two reasons why dynamic programming is a good approach to solving this problem

- *Divides the problem up into small computationally tractable steps. Only has to store the immediately preceding values, not all of the info*
- *Can find the path by keeping track of just the backtrace*

Question 4

4. a. What is the difference between a Markov Chain and a Hidden Markov Model. What's "Hidden" in the HMM.
   - *In a Markov chain, the path is completely determined by the input. There is no ambiguity*
   - *In an HMM, the states are "hidden". The same sequence input can result in different state sequences.*

4. b Give an example of a problem that can be solved with a Markov chain and one that requires an HMM.
   - *Markov chain: FSA to accept or reject sentences and give their likelihood*
   - *HMM: Part of speech tagging*

Question 5

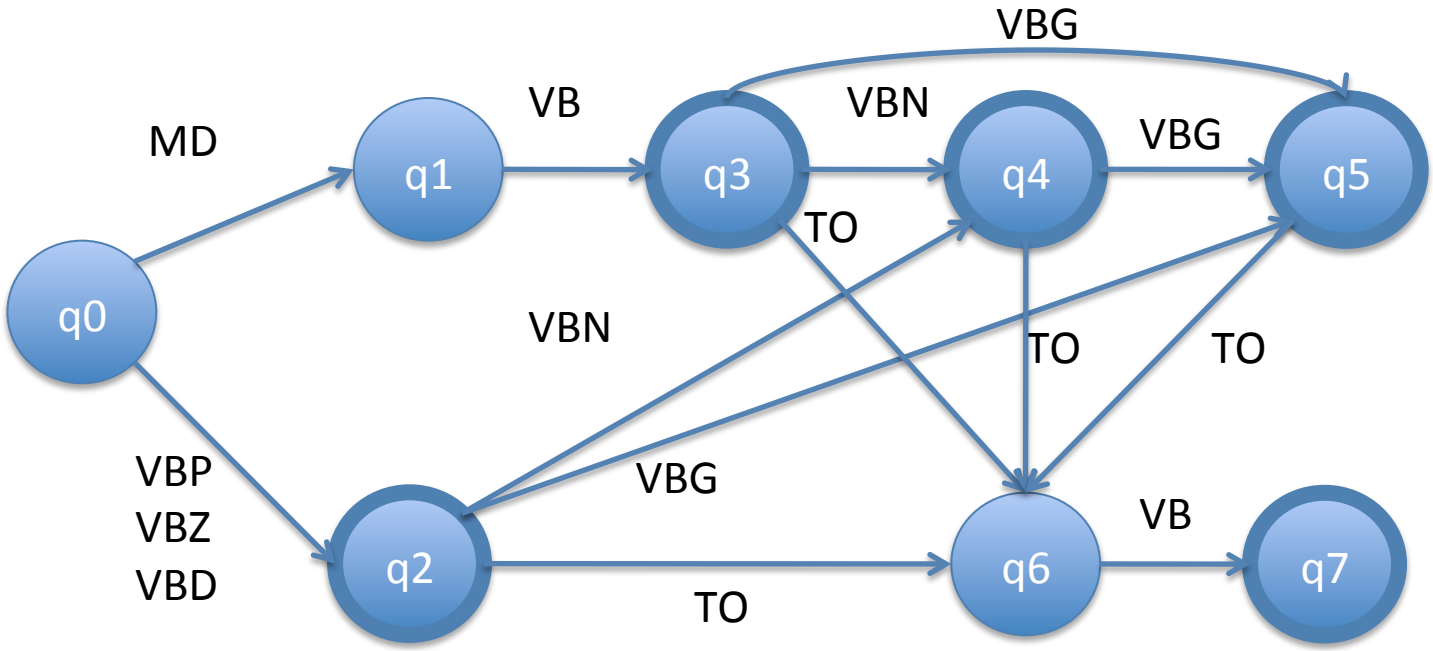
- 4.a. What is an ngram model? How can ngram models be computed?
  - *Ngram model is a computation of a seqence (e.g. words) where the likelihood of one element in the sequence is based on the previous N-1 elements.*
- 4.b. What is smoothing and why do we need to do it?
  - *Smoothing allows us to model events that have not been seen by assigning them a small probability and then ensuring the probabilities still sum to one*
  - *Particularly in text processing, there will always to events that have not occurred in the corpus. (and in general math works out better without 0's)*

Question 5

5c. Briefly describe backoff and interpolation

- Backoff is using the probability from an event that requires less information, such as when there is no data for a trigram, use the bigram
- Interpolation is a method of combining multiple models by using a weight for each model so that the probabilities still add to 1

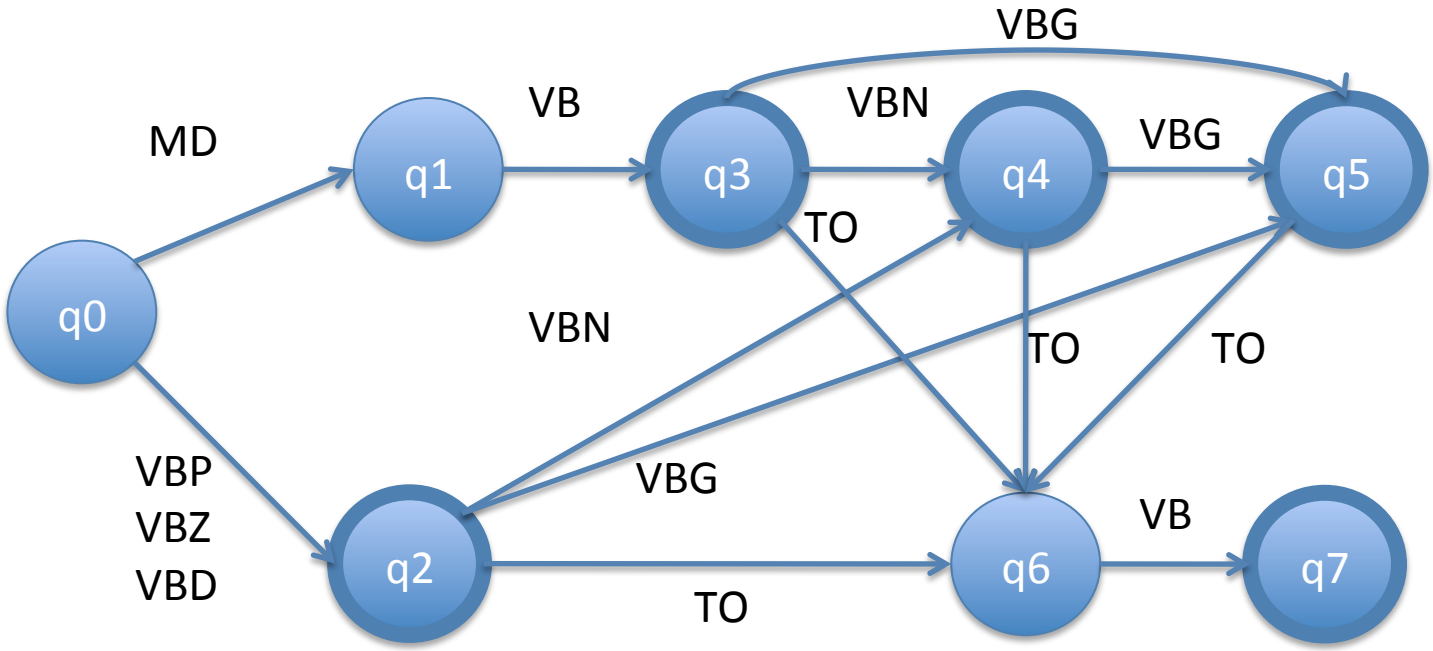
6. Verb Group FSA (16 pts)



I **could have danced** all night: MD VB VBN
I **was dancing** when the lights went out: VBD VBG
We **danced** the night away: VBD
I **would have been dancing**, but …: MD VB VBN VBG

He **has danced** his whole life: VBZ VBN
She **dances** four times a week: VBZ
He **loves to dance:** VBZ TO VB
She **might dance** with him: MD VB

6. Verb Group FSA (16 pts)



I **could have danced** all night: MD VB VBN
I **was dancing** when the lights went out: VBD VBG
We **danced** the night away: VBD
I **would have been dancing**, but …: MD VB VBN VBG

He **has danced** his whole life: VBZ VBN
She **dances** four times a week: VBZ
He **loves to dance:** VBZ TO VB
She **might dance** with him: MD VB

6 Verb FSA

- Embed this FSA within a sentence FSA.

6a (2)

- Describe the algorithm you would use
  - For each word in the input, look up the categories
    - Select the first category, push the rest on a backup stack
    - If the category is allowed from the current state, make that transition
      - Else if there is a category on the backup stack check that state (and pop the stack)
    - If the input is at the end and you are in a final state, accept
    - Else, if there is additional input, repeat
    - If there are no transitions you can make and no categories on the backup stack, fail

Question 6 B

- Illustrate your algorithm
  - This had to show backup!! No "omniscient" algorithms
- Extra credit
  - What's wrong with "They may work having"
  - (bad example, since due to ambiguity of "may")
  - The verb group requires not just POS, but particular VERBS
  - "They may be working" is fine. FSA has to stipulate particular verbs, not just POS

Question 7

a. Now you want to create an HMM of the verb group in English. What does the transition table look like? (just describe it) and what other information do you need?

- *You need transition probabilities (from counting POS sequences in a corpus) and observation probabilities (p(word|tag)).*

b. What are the steps you would need to create an HMM to solve this problem?

- *Annotate data, build dictionary, count ngrams, create probability model, select smoothing, backoff, interpolations models*

Question 8

Using a "multiple tape" finite state transducer for morphology is an example of a "cascaded architecture".

a Consider the problem of going from a base form with features (e.g. dance+VB+gerund) to the correct surface spelling ("dancing"). Describe the steps of a multiple tape FST (just characterize the main actions, you don't have to write it all out).

- *The first step takes the features and turns them into the default affixes and inserts the morpheme boundaries*
- *The second step transforms that into the surface spelling, taking into consideration spelling rules like dropping final e's and doubling consonants.*

Question 8b

Why is this cascaded architecture useful here? Give an example of another computational linguistic problem where it would be useful

- *By breaking the problem into multiple steps, the problem is easier to solve and can make use of generalities (gerund => "ing") in step 1 and spelling rules apply to many different affixes.*
- *Many problems are capable of being broken into a series of smaller steps. For example, first determining POS, which reduces ambiguity, and then parsing.*

Question 9

A High school teacher assigned her class to write biographies of famous 20th century scientists, but after reading them, suspected that they were plagiarized out of Wikipedia and other online sources she had pointed them for the research. Describe how you could use techniques learned so far in this class to help her figure this out who plagiarized?

- *Build an ngram model for each of the various sources. Test the students data against each and measure the perplexity. A low perplexity indicates a high likelihood of plagiarism.*

Question 10

You are building an ngram model of a corpus. Should you stem the words and do the counts or leave them in the surface form? Give pros and cons and include what characteristics of the corpus might influence your decision.

- *Stemming the words means there will be fewer types, since there will just be base forms. This means that some generalizations will be captured (He swam, he swims ..> He swim). However, there are some generalization that won't be captured (I swim vs. she swims),*
- *This is a good idea when there is a small amount of data and there are fewer examples of the ngrams or in highly inflected languages where there are many different forms of each word.*
- *If a large amount of data available, however, ngrams over the surface forms can be more powerful and precise.*