CS4740/CS5740

Introduction to Natural Language Processing

2016 Midterm Solutions

1 Ambiguity in NLP (25 points)

Consider the following sentence:

I wanted to plant some trees in our yard to block the view of our neighbors' driveway, but they were too expensive.

1. There are a number of **part-of-speech** ambiguities that affect the interpretation of the sentence above.

   (a) (5 pts) Describe one such instance of a part-of-speech ambiguity that involves a **function word** (i.e. a **closed class** word) from the sentence.

   Solution: "to" in "to plant" or "to block" can be a preposition ("go to the store") or part of an infinitival verb phrase (i.e. it introduces an infinitive, as in both instances of "to" in the example sentence).

   "some" can be a determiner ("some trees"), a prounon ("here are some of the options") or an adverb ("the rules can change some").

   "but" can be a conjunction (as in the example sentence), a noun ("no but(s) about it"), a preposition ("nothing but air"), an adverb ("he is but a child").

   Other options: "too", "in"

   2 pts: for selecting one of the above function words.
   3 pts: for giving both the correct part of speech and one of the alternatives **with** an example.
   -1 pt: for no example or description using the alternative POS.

   (b) (5 pts) Describe a second part-of-speech ambiguity in the sentence, this time one that involves a **content word**.

   Solution: "plant" can be a verb (as in the example) or a noun ("Claire's plant needs watering".

   "block" can be a verb (as in the example) or a noun ("put the block in the bag").

   "tree" can be a noun (as in the example) or a verb ("Marseille treed the cat")

   There are other options as well: view (noun, verb), neighbor (noun, verb).

2 pts: for selecting one of the above content words.
3 pts: for giving both the correct part of speech and one of the alternatives **with** an example.
-1 pt: for no example or description using the alternative POS.

2. (5 pts) Describe one **discourse-level** ambiguity that occurs in the sentence.

   Solution: "they" can refer to the neighbors or the trees (or any plural entity in the discourse).

   5 pts: for giving the solution above.
   2 pts: for knowing that a "discourse-level" ambiguity typically involves more than one sentence or more than one clause, but not giving the pronoun resolution solution.

3. (5 pts) Consider the task of constructing a lexicon to be used for an arbitrary NLP task in English. Show what the **lexeme(s)** for the **noun** *block* might look like.

   Solution:

   block | *buh-lok* or some phonetic transcription | a large solid piece of material

   5 pts: lexicon shows one entry: "block", a pronunciation, and a meaning.
   3 pts total: if missing ONE component of the lexeme (spelling/word; pronunciation; or meaning)
   2 pts total: if missing TWO components (spelling/word; pronunciation; meaning) of each of the lexemes
   +3 (bonus!) pts: if a second **related** definition is included for the noun lexeme. For the noun lexeme, one possibility for this is the meaning of "block" that refers to a city block.

4. (5 pts) What does it mean to say that *Zipf's Law applies to verbs and their word senses*?

   Solution: A very small number of verbs have many word senses while the vast vast majority of verbs have very few (1 or 2) word senses.

   5 pts: For something along the lines of the solution above.
   2 pts: If the solution just states Zipf's Law (correctly, i.e. frequency is proportional to 1/rank) but not in terms of verbs and word senses. 1 pt if it is stated somewhat correctly.

2 Word Sense Disambiguation and Evaluation (25 points)

	POS	sense-id	
time	n	5	(the continuum of experience in which events pass from the future through the present to the past)
time	v	1	(measure the time or duration of an event or action or the person who performs an action in a certain period of time)
flies	n	1	(two-winged insects characterized by active flight)
flies	v	8	(pass away rapidly)
like	v	4	(feel about or towards; consider, evaluate, or regard)
like	a	1	(resembling or similar; having the same or some of the same characteristics; often used in combination)
arrow	n	1	(a mark to indicate a direction or relation)
arrow	n	2	(a projectile with a straight thin shaft and an arrowhead on one end and stabilizing vanes on the other; intended to be shot from a bow)

1. (15 pts) Assume the subset of WordNet senses provided above. Simulate the **Lesk word sense disambiguation algorithm** for the word **flies** given the context: *Time* **flies** *like an arrow.*

   Be sure to show the score the algorithm would assign to each of the senses for the target word. For the computations, assume that no part-of-speech disambiguation has been provided. State any additional assumptions needed.

   Solution:

   Disambiguating flies:

   (**10 pts**) for making clear via your simulation that you know the dictionary-based WSD algorithm of Lesk; **otherwise**

   (**5 pts**) if the Lesk algorithm is described but no simulation is provided for the example.

   For the remaining 5 pts:

   flies-n-1 shares no content words with any of the context words — *time*, *like* or *arrow*. So a score of 0. (**3pts**)

   flies-v-8 shares *pass* with time-n-5 for a score of 1. (**2 pts**)

   So the algorithm selects flies-v-8.

2. (10 pts) Suppose that two WSD systems are evaluated on a test set of five examples and that the target word to be disambiguated in each example has two possible word senses, *s1* and *s2*. The following table shows the output for each system as well as the desired (gold standard) answers:

	WSD system 1	WSD system 2	Answer key
1	s1	s1	s1
2	s2	s2	s2
3	s1		s2
4	s1		s2
5	s1	s1	s1

Compute the **Precision**, **Recall** and **F-measure (F1 score)** for each WSD system.

Solution:

2pts for having one of P or R correct; **+1pt** if both P and R are correct; **+1** if F-score is also correct:
WSD system 1: P = 3/5. R = 3/5. F = 2PR/(P+R) = 3/5.
(**3+1+2pts**:) WSD system 2: P = 3/3. R = 3/5. F = 2PR/ (P+R) = (2 x 0.6) / 1+0.6 = 1.2/1.6 = 3/4 = 0.75.

3 Language Modeling (30 points)

I like traffic lights , I like traffic lights , I like traffic lights , no matter where they 've been . I like traffic lights , I like traffic lights , I like traffic lights , I like traffic lights , I like traffic lights , but only when they 're green .

Assume that the above text (actual lyrics from the Monty Python song, *I Like Traffic Lights*) is provided as the (entire) training corpus for a **bigram language model**. For preprocessing: assume that **all words are converted to lower case**; do not add beginning (or end) of sentence markers. No unknown word handling is required.

1. (7 pts) Using **Maximum Likelihood Estimation** and the **bigram model** derived from the above training data, compute *P(only green traffic lights)*.

   Solution:

   There are 17 word types. (Some will miss the 've and the 're and so will get a count of 15. This is ok.)

   P_{MLE}(only green traffic lights) =
   (**7 pts**) for the answer of 0 (since I forgot to ask that you show your work); otherwise,

   (**4 pts**) P_{MLE}(green|only) x P_{MLE}(traffic|green) x P_{MLE}(lights|traffic) =
   (**3 pts**) 0/1 x 0/1 x 8/8 = 0 (one point for each correct term)

   It's ok if students tried to put in a start and/or end state (or tried to do *something* to handle this). Just ignore those terms.

2. (13 pts) Now using **add-one (Laplacian) smoothing** and the **bigram model** derived from the above training data, compute the same thing: *P(only green traffic lights)*.

   Solution:

   P_{LAP}(only green traffic lights) =

   (**4 pts**) P_{LAP}(green|only) $= (0+1)/(1+17) = 1/18$

   (**3 pts**) P_{LAP}(traffic|green) $= (0+1)/(1+17) = 1/18$

   (**4 pts**) P_{LAP}(lights|traffic) $= (8+1)/(8+17) = 9/25$

   (**2 pts**) So P_{LAP}(only green traffic lights) = 1/18 x 1/18 x 9/25

3. (4 pts) How many unseen bigrams are there for the *I Like Traffic Lights* corpus?

   Solution: 17 word types. So 17 x 17 = 289 possible bigrams. 19 bigrams appeared in the corpus, leaving 289 - 19 = 270 unseen bigrams.

   1pt for getting the number of possible bigrams correct; **2pts** for getting the number of seen bigrams correct; **1 pt** for subtracting the second from the first.

4. (6 pts) Given a test corpus $W = w_1 w_2 w_3 \ldots w_N$, show the formula (i.e. equation) for computing the **perplexity** of a **trigram language model** (trained on some corpus) on W. (Note: this question has nothing to do with the *I Like Traffic Lights* corpus. We do not expect to see any actual perplexity scores.)

   Solution:

   (**3pts** if only this is provided:) $\mathrm{PP}(W) = P(w_1 w_2 w_3 \ldots w_N)^{-1/N}$
   (**6pts** if the trigram version is worked out) =

   the Nth root of $1/(P(w_3|w_1 w_2)P(w_4|w_2 w_3)P(w_5|w_3 w_4)\ldots P(w_N|w_{N-2}w_{N-1}))$