Försättsblad till skriftlig tentamen vid Linköpings universitet

Datum för tentamen	2019-03-18
Sal (1)	KÅRA(36)
Tid	14-18
Utb. kod	729G17
Modul	TEN3
Utb. kodnamn/benämning **Modulnamn/benämning**	Språkteknologi Skriftlig tentamen
Institution	IDA
Antal uppgifter som ingår i tentamen	8
Jour/Kursansvarig Ange vem som besöker salen	Marco Kuhlmann
Telefon under skrivtiden	013-284644
Besöker salen ca klockan	15
Kursadministratör/kontaktperson (namn + tfnr + mailaddress)	Veronica Kindeland Gunnarsson, 013-285634, veronica.kindeland.gunnarsson@liu.se
Tillåtna hjälpmedel	inga
Övrigt	
Antal exemplar i påsen	

Försättsblad till skriftlig tentamen vid Linköpings universitet

Datum för tentamen	2019-03-18
Sal (1)	KÅRA(12)
Tid	14-18
Utb. kod	TDP030
Modul	TEN1
Utb. kodnamn/benämning **Modulnamn/benämning**	Språkteknologi Skriftlig tentamen
Institution	IDA
Antal uppgifter som ingår i tentamen	8
Jour/Kursansvarig Ange vem som besöker salen	Marco Kuhlmann
Telefon under skrivtiden	013-284644
Besöker salen ca klockan	15
Kursadministratör/kontaktperson (namn + tfnr + mailaddress)	Veronica Kindeland Gunnarsson, 013-285634, veronica.kindeland.gunnarsson@liu.se
Tillåtna hjälpmedel	inga
Övrigt	
Antal exemplar i påsen	

Exam 2019-03-18

Marco Kuhlmann

This exam consists of two parts:

Part A consists of 5 items, each worth 3 points. These items test your understanding of the basic algorithms that are covered in the course. They require only compact answers, such as a short text, calculation, or diagram.

Part B consists of 3 items, each worth 6 points. These items test your understanding of the more advanced algorithms that are covered in the course. They require detailed and coherent answers with correct terminology.

Note that surplus points in one part do not raise your score in another part.

Grade requirements 729G17/729G34:

- Grade G: at least 12 points in Part A
- Grade VG: at least 12 points in Part A and at least 14 points in Part B

Grade requirements TDP030:

- Grade 3: at least 12 points in Part A
- Grade 4: at least 12 points in Part A and at least 7 points in Part B
- Grade 5: at least 12 points in Part A and at least 14 points in Part B

Wildcards: When grading the exam, we will credit you with the maximal number of points from up to three wildcards. Wildcards are only valid for Part A. The numbering of the questions corresponds to the numbering of the wildcards.

Good luck!

Part A

Note: When instructed to 'answer with a fraction', you should provide a fraction containing concrete numbers and standard mathematical operations, but no other symbols. You do not need to simplify the fraction.

like this: $\frac{42+13}{100}$ not like this: $\frac{\#(a)+\#(b)}{N}$

01 Text classification (3 points)

a) A certain text classifier has to decide whether incoming news items are news about China (class `C`) or news about Japan (class `J`). Describe a situation where accuracy is an ill-chosen evaluation measure for such a classifier.

b) Use Maximum Likelihood estimation (without smoothing) to estimate the class probabilities and word probabilities of a Naive Bayes text classifier on the following document collection. Answer with fractions.

	document	class
1	Chinese Beijing Chinese	`C`
2	Chinese Chinese Shanghai	`C`
3	Chinese Macao	`C`
4	Tokyo Japan Chinese	`J`

c) Based on the probabilities just estimated, which class does the classifier predict for the document 'Chinese Chinese Chinese Tokyo Japan'? Show that you have understood the Naive Bayes classification rule.

02 **Language modelling** **(3 points)**

The Corpus of Contemporary American English (COCA) is the largest freely-available corpus of English, containing approximately 560 million tokens. In this corpus we have the following counts of unigrams and bigrams:

snow	*white*	*white snow*	*purple*	*purple snow*
38,186	256,091	122	11,218	0

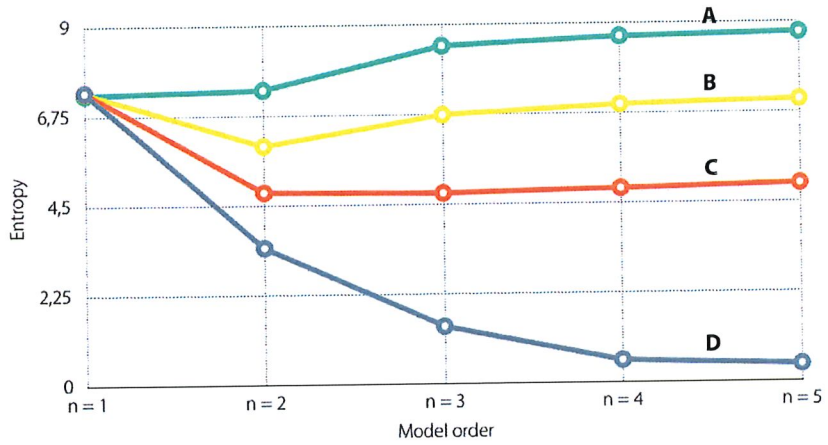
a) Estimate the following probabilities using maximum likelihood estimation without smoothing. Answer with fractions.

i. $P(snow)$ ii. $P(snow \mid white)$

b) Estimate the following probabilities using maximum likelihood estimation with additive smoothing, $k = 0.01$. Assume that the vocabulary consists of 1,254,100 unique words. Answer with fractions.

i. $P(snow)$ ii. $P(snow \mid purple)$

c) We use maximum likelihood estimation with add-k smoothing to train n-gram models on the COCA corpus, with $n \in \{1, \ldots, 5\}$ and $k \in \{0, 0.01, 0.1, 1\}$. The following graph shows the entropy of each trained model on the training data. Which series corresponds to which k-value, and why? Answer with a short text.



03 Part-of-speech tagging (3 points)

a) The evaluation of a part-of-speech tagger produced the following confusion matrix. The marked cell gives the number of times the system classified a document as class NN whereas the gold-standard class for the document was JJ.

	NN	JJ	VB
NN	58	6	1
JJ	5	11	2
VB	0	7	43

Set up fractions for the following values:

i. accuracy

ii. precision with respect to class JJ

b) The following matrices specify (parts of) a hidden Markov model. The marked cell specifies the probability for the transition from BOS to AB.

	AB	PN	PP	VB	EOS
BOS	1/11	1/10	1/12	1/11	1/25
AB	1/11	1/11	1/11	1/10	1/14
PN	1/11	1/12	1/12	1/10	1/16
PP	1/13	1/11	1/12	1/14	1/18
VB	1/11	1/10	1/10	1/13	1/15

	she	got	up
AB	1/25	1/25	1/14
PN	1/13	1/25	1/25
PP	1/25	1/25	1/13
VB	1/25	1/14	1/19

Provide a fraction for the probability of the following tagged sentence:

she	got	up
PN	VB	PP

c) Which of the following features of a multi-class perceptron tagger go beyond what can be expressed in a hidden Markov model? Answer with a short text.

1 current word

2 word to the left of the current word

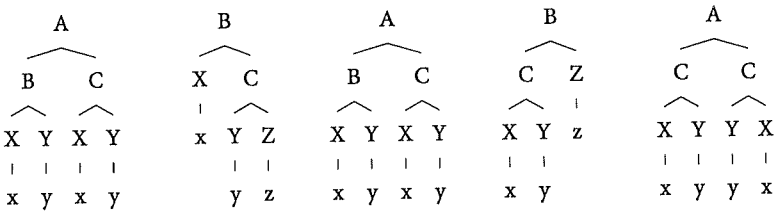
3 word to the right of the current word

4 part-of-speech tag of the word to the left of the current word

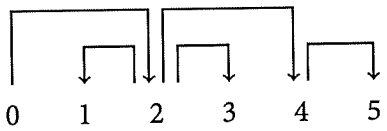
04 **Syntactic analysis** **(3 points)**

a) You add up the probabilities of all rules of a certain probabilistic context-free grammar and get the number 342.3. Explain why there must be an error in your calculation.

b) Below is a small phrase structure treebank. Read off all rules whose left-hand sides are either B or C and estimate their rule probabilities using maximum likelihood estimation (no smoothing).



c) State two different sequences of transitions that make the transition-based dependency parser produce the following dependency tree:



05 **Semantic analysis** **(3 points)**

a) Draw a partial WordNet-hierarchy for the following synsets:

1 university
2 institution, establishment
3 educational institution
4 hospital
5 kindergarten
6 medical institution

b) Here are three signatures (glosses and examples) from Wiktionary for different senses of the word *course*:

1 A normal or customary sequence. **2** A learning program, as in university. *I need to take a French course.* **3** The direction of movement of a vessel at any given moment. *The ship changed its course 15 degrees towards south.*

Based on these signatures, which of the three senses of the word *course* does the Lesk algorithm predict in the following senses? Ignore the word *course*, stop words, and punctuation.

In the United States, the normal length of a course is one academic term.

c) In a certain set of word vectors derived from a co-occurrence matrix, the two nearest neighbours of the word *fast* are *quick* and *slow*. Explain why this is not an unexpected result. Answer with a short text.

Part B

06 **Minimum edit distance** **(6 points)**

a) Define the concept of the Levenshtein distance between two words. The definition should be understandable even to readers who have not taken this course.

b) Compute the Levenshtein distance between the two words *leda* and *deal* using the Wagner–Fischer algorithm. Your answer should contain both the distance itself and the complete matrix.

c) It is much more likely for a user to mistype the word *deal* as *seal* than as *beal*; this is because the keys for the letters *d* and *s* are much closer to each other on the keyboard than the keys for the letters *d* and *b*. Explain how the Wagner–Fischer algorithm could be adapted to take this information into account.

07 **Viterbi algorithm** **(6 points)**

Here is a Hidden Markov model (HMM) specified in terms of costs (negative log probabilities). The marked cell gives the transition cost from BOS to PL.

	PL	PN	PP	VB	EOS
BOS	11	2	3	4	19
PL	17	3	2	5	7
PN	5	4	3	1	8
PP	12	4	6	7	9
VB	3	2	3	3	7

	they	freak	out
PL	17	17	4
PN	3	19	19
PP	19	19	3
VB	19	8	19

When using the Viterbi algorithm to calculate the least expensive (most probable) tag sequence for the sentence 'they freak out' according to this model, one gets the following matrix. Note that the matrix is missing some values.

		they	freak	out
BOS	0			
PL		28	27	
PN		5	28	
PP		22	27	
VB		23	14	
EOS				

a) Calculate the missing values (the last two columns).

b) Let m and n denote the number of tags in the HMM and the number of words in the input sentence, respectively. The memory required by the Viterbi algorithm is in $O(mn)$, and the runtime required is in $O(m^2n)$. Explain what these statements mean and how to derive them.

c) When one is only interested in the *cost* of the least expensive tag sequence, not in the sequence itself, then the memory required by the Viterbi algorithm is in $O(m)$. Explain this statement. Why does this statement not hold if one wants to reconstruct the actual tag sequence?

08 Named entity tagging (6 points)

Named entity tagging is the task of identifying entities such as persons, organisations, and locations in running text. One way to approach this task is to use the same techniques as in part-of-speech tagging. However, a complicating factor is that named entities can span more than one word. Consider the following sentence:

Alfred Nobel was an inventor from Sweden.

In this example, while the unigram 'Sweden' corresponds to one named entity of type 'location' (LOC), we would also like to identify the bigram 'Alfred Nobel' as a mention of *one* named entity, of type 'person' (PER).

a) To account for the fact that named entities can span more than one word, we can use the so-called IOB tagging. Explain how this scheme works and show how the example sentence given above would be tagged using IOB tagging.

b) Named entity taggers using the IOB tagging scheme can be evaluated at the level of words or at the level of entities. Explain what this means and provide a concrete example which shows that a system can have a high word-level accuracy but a low entity-level accuracy. Your example should contain at least one named entity of type person (PER).

c) Named entity taggers often use *gazetteers*. Explain what a gazetteer is and how it can be integrated into a named entity tagger based on the multi-class perceptron.