Exam 2017-08-25

Marco Kuhlmann

This exam consists of three parts:

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

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

3. **Part C** consists of 1 item worth 12 points. This item is an essay question that tests your understanding of the following article:

   David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A. Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John Prager, Nico Schlaefer, Chris Welty. *Building Watson: An Overview of the DeepQA Project*. AI Magazine 31(3):59–79, 2010.

4. **De två extrauppgifterna i slutet på tentan ska du endast bearbeta om din förstagångsregistrering gäller den gamla provuppsättningen där tentan hade provkoden TEN2 – t.ex. ifall du blev förstagångsregistrerad år 2016.**

Grade requirements: For grade G (Pass), you need at least 12 points in Part A. For grade VG (Pass with distinction), you additionally need at least 12 points in Part B, and at least 6 points in Part C.

Om du skriver tentamen enligt provkoden TEN2 så behöver du sammanlagt minst 15 poäng från del A (Part A) och de två extrauppgifter i slutet av tentan för att bli godkänd på tentan (betyg G). För VG gäller samma krav som ovan.

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

Good luck!

Part A

01 Text classification (3 points)

A Naive Bayes classifier has to decide whether the document 'London Paris' is news about the United Kingdom (class `U`) or news about Spain (klass `S`).

a) State the formula for the Naive Bayes classification rule and explain its parts.

b) Estimate the probabilities that are relevant for the classification of the specific document above from the following document collection using Maximum Likelihood estimation (without smoothing). Answer with fractions.

	document	class
1	London Paris	`U`
2	Madrid London	`S`
3	London Madrid	`U`
4	Madrid Paris	`S`

c) Practical implementations of a Naive Bayes classifier often use log probabilities. Draw a graph for the function $f(p) = \log p$, where p is a probability. Explain how the classification rule needs to be changed when log probabilities are used.

02 Language modelling (3 points)

The 520 million word Corpus of Contemporary American English contains 1,254,193 unique words. We have the following counts of unigrams and bigrams: *your*, 883,614; *rights*, 80,891; *doorposts*, 21; *your rights*, 378; *your doorposts*, 0.

a) Estimate the probabilities $P(\textit{rights})$ and $P(\textit{rights} \mid \textit{your})$ using Maximum Likelihood estimation (no smoothing). Answer with fractions.

b) Estimate the bigram probability $P(\textit{doorposts} \mid \textit{your})$ using Maximum Likelihood estimation and add-k smoothing with $k = 0.1$. Answer with a fraction.

c) Suppose that we train three n-gram models on 38 million words of newspaper text: a unigram model, a bigram model, and a trigram model. Suppose further that we evaluate the trained models on 1.5 million words of text with the same vocabulary and obtain the following entropy scores: 7.409, 6.768, and 9.910. Which entropy belongs to which model? Provide a short explanation.

03 Part-of-speech tagging (3 points)

The evaluation of a part-of-speech tagger produced the following confusion matrix. The marked cell gives the number of times the system tagged a word as a verb (`VB`) whereas the gold standard specified it as a noun (`NN`).

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

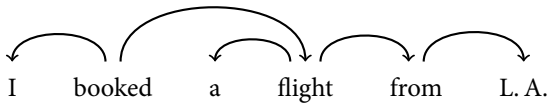
a) Set up a fraction for the tagger's accuracy.

b) Set up fractions for the tagger's recall on adjectives and its precision on nouns.

c) Write down another confusion matrix where accuracy is the same as in the matrix above, but where the tagger's recall on verbs is 100%.

04 Syntactic analysis (3 points)

A transition-based dependency parser analyses the sentence *I booked a flight from L. A.* Here is the gold-standard tree for this sentence.



a) Suppose that the parser starts in the initial configuration for the sentence and takes the transitions `SH`, `SH`, `RA`. State the new configuration. Represent the partial dependency tree by listing its arcs.

b) State a complete sequence of transitions that takes the parser all the way from the initial configuration to a terminal configuration, and that recreates all arcs of the gold-standard tree.

c) For a sentence with n words, how many transitions does the parser make from the initial configuration to a terminal configuration? Explain your reasoning.

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

The Lesk algorithm is a simple method for word sense disambiguation that relies on the use of dictionaries. 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.*

a) Which of the three senses of the word *course* does the Lesk algorithm predict in the following sentence, based on the given signatures?

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

Ignore the word *course*, punctuation, and stop words. Explain your answer.

b) Change the sentence such that the word *course* maintains its intended sense, but the Lesk algorithm now predicts a different sense than in the previous item.

c) An alternative to ignoring stop words is to use the extension of the Lesk algorithm known as Corpus Lesk, in which each word w is weighted as follows:

$$\text{weight}(w) = \log \frac{\text{number of signatures}}{\text{number of signatures which contain } w}$$

Under this scheme, what can we you about the weight of a stop word, assuming a sufficiently large corpus of signatures with many examples? Explain.

Part B

06 **Levenshtein distance** **(6 points)**

The following matrix shows the values computed by the Wagner–Fisher algorithm for finding the Levenshtein distance between the two words *intention* and *execution*. Note that the matrix is missing some values (marked cells).

n	A	8	8	8	8	8	8	7	6	5
o	A	7	7	7	7	7	7	6	5	6
i	A	6	6	6	6	6	6	5	6	7
t	A	5	5	5	5	5	5	6	7	8
n	A	4	4	4	4	5	6	7	7	7
e	A	3	4	B	4	5	6	6	7	8
t	A	3	3	3	4	5	6	6	7	8
n	A	2	2	3	4	5	6	7	7	7
i	A	1	2	3	4	5	6	6	7	8
#	A	A	A	A	A	A	A	A	A	A
	#	e	x	e	c	u	t	i	o	n

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) Provide the values for the cells marked A. Explain.

c) Calculate the value for the cell marked B. Explain. Show that you have understood the Wagner–Fisher algorithm.

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

The following matrices specify a Hidden Markov model 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

	hen	vilar	ut
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 'hen vilar ut' according to this model, one gets the following matrix. Note that the matrix is missing three values (marked cells).

		hen	vilar	ut
BOS	0			
PL		28	27	21
PN		5	B	35
PP		22	27	20
VB		A	14	36
EOS				C

a) Calculate the value for the cell A. Explain your calculation.

b) Calculate the values for the cells B and C. Explain your calculations.

c) Starting in cell C, draw the backpointers that identify the least expensive (most probable) tag sequence for the sentence. State that 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 idea 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:

Thomas Edison was an inventor from the United States.

In this example we would like to identify the bigram 'Thomas Edison' as a mention of *one* named entity of type 'person' (PER), not two words; similarly, we would like to identify 'United States' as one named entity of type 'location' (LOC).

To solve this problem, we can use the so-called BIO tagging. In this scheme we introduce a special 'part-of-speech' tag for the beginning (B) and inside (I) of each entity type, as well as one tag for words outside (O) any entity. Here is the example sentence represented with BIO tags:

$\text{Thomas}_{\text{B-PER}}$ $\text{Edison}_{\text{I-PER}}$ was_{O} an_{O} $\text{inventor}_{\text{O}}$ from_{O} the_{O} $\text{United}_{\text{B-LOC}}$ $\text{States}_{\text{I-LOC}}$

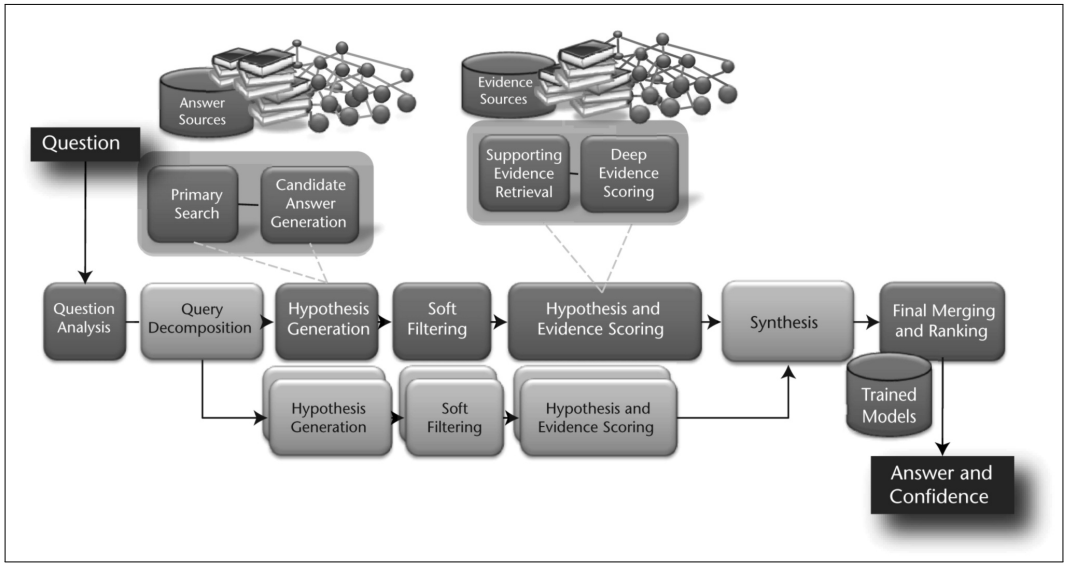
a) Explain what type of data would be required for training named entity taggers that use the BIO tagging scheme using supervised machine learning.

b) Discuss which type of gold-standard data it is easier to obtain more of: data for part-of-speech taggers or data for named entity taggers.

c) Named entity taggers using the BIO tagging scheme can be evaluated at the level of words ('How many words were tagged with the correct BIO tag?') or at the level of entities ('How many *n*-grams were tagged with their correct entity type?'). Provide a concrete example, similar to the one above, which shows that a system can have a high word-level tagging accuracy but a low entity-level tagging accuracy. Your example should contain at least one named entity of type person (PER). Explain.

Part C

09 **Question answering systems** **(12 points)**



a) Using the diagram as a point of reference, explain the DeepQA architecture from a bird's eye perspective. Provide examples from the article.

b) Explain one of the components in the DeepQA architecture in more detail. Provide examples from the article.

c) Describe the measures that the developers used to evaluate Watson. According to your own judgement, which of those measures is the most important?

d) The authors write: 'Our results strongly suggest that DeepQA is an effective an extensible architecture that may be used as a foundation for combining, deploying, evaluating, and advancing a wide range of algorithmic techniques to rapidly advance the field of QA.' Discuss which specific results in the article support this conclusion. Would you agree with the authors' assessment?

Extra uppgifter TEN2

Dessa uppgifter ska du endast bearbeta om din förstagångsregistrering gäller den gamla provuppsättningen där tentan hade provkoden TEN2 – t.ex. ifall du blev förstagångsregistrerad år 2016.

10 **Semantisk analys** **(3 poäng)**

Betrakta följande (normaliserade) dokumentsamling:

(1) automobile wheel motor vehicle transport passenger

(2) car form transport wheel capacity carry five passenger

(3) transport London game spectator advise avoid use car

(4) London soccer tournament begin goal match

(5) Giggs score goal football tournament Wembley London

(6) Bellamy passenger football match play part goal

a) Komplettera följande term–term matris utifrån dokumentsamlingen.

	passenger	transport	goal	match
automobile	☐	☐	☐	☐
car	☐	☐	☐	☐
soccer	☐	☐	☐	☐
football	☐	☐	☐	☐

b) Rita in målorden som vektorer i ett koordinatsystem där x-axeln svarar mot det totala antalet förekomster i kontexten *passenger*, *transport* och y-axeln svarar mot det totala antalet förekomster i kontexten *goal*, *match*.

c) Förklara hur man med hjälp av sådana vektorrepresentationer kan mäta likhet mellan målorden. Vilka resultat skulle denna metod ge för de angivna målorden?

11 **Frågebesvarande system** **(3 poäng)**

Följande frågor ställdes till ett frågebesvarande system:

A Vilket datum förliste *S/S Per Brahe*?

B Hur var vädret den dagen?

C Var ligger varvet där *S/S Per Brahe* byggdes?

a) Ange svarstyper för de tre frågorna.

b) Olika frågor är olika svåra för frågebesvarande system att hitta rätt svar på. Rangordna de tre frågorna från lättast till svårast. Motivera din rangordning. Antag att systemet inte har någon egen kunskapsdatabas utan bygger på dokumentsökning i svenska Wikipedia.

c) Ferrucci et al. (2010) undersökte ett slumpmässigt urval av 20 000 Jeopardy-frågor och hittade 2 500 distinkta svarstyper. De fann att de 200 mest frekventa svarstyperna täckte mindre an 50% av fragorna. Förklara varför detta är ett problem för frågebesvarande system som använder sig av maskininlärning.