BBM 495
Introduction to Natural Language Processing
Midterm 2 – 21.05.2020 (Duration: 180 min)

Name-Surname:
Student no:

I attest that I have not given or received aid in this examination.

Signature:

Part I: Short Answers (each 4 points):

a) State one drawback of using chain rule. What is your solution to tackle with this drawback in language models?

Sparse probabilities is one drawback.

Markov assumption is one possible solution.

b) State one inadequacy of language models that we discussed in the class. Give an example in English. What is your solution to tackle with this inadequacy?

Recursions cannot be handled and language models always depend on only a limited length of history.

The solution is to use finite state automata, or context free grammars, which will never forget the previous states.

An example in English: the book that I read when I was in high school was titled ...

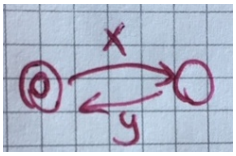
Here the end of the word will be dependent on the word book, which requires a long term dependency.

Using Markov assumption (n-grams) is not a remedy for this problem because n-gram also refers to a language model. Using smoothing is not a correct answer because smoothing is naturally applied for language models and it is called smoothed language model.

c) We already learned that every regular expression can be converted into a finite state automaton. Prove it with an example along with a conversion for a language with two letters in the alphabet: {x, y}.

xyxyxyxy
(xy)*

The corresponding FSA also should be drawn along with this regular expression.



d) Give at least two methods that can be used for estimating the semantic similarity between words? Only the ones learned in the class will be graded. Compare their advantages and disadvantages.

Thesaurus-based methods
Co-occurrence matrixes - distributional methods
Neural methods - word embeddings

Thesaurus-based methods require human effort to be updated and manually constructed. Neural methods do not require any annotation, just raw corpus is sufficient. Co-occurrence matrices could be very high-dimensional and sparse, so they have some computational problems.

e) Describe what "You shall know a word by the company it keeps" (Firth, 1957) means and give an example in English to support your description.

The meaning/sense of a word can be inferred from its neighbour words.
Distributional hypothesis

Example:
The xxx was written by the same author.
The author wrote this xxx in 1924.

The same example on the slide is not graded! You should give your own example.

f) How is the conditional probability between two words is predicted by word2vec (Mikolov et al. 2013)? Describe mathematically by giving what each symbol refers to in your equation. Giving only the equation will not be graded.

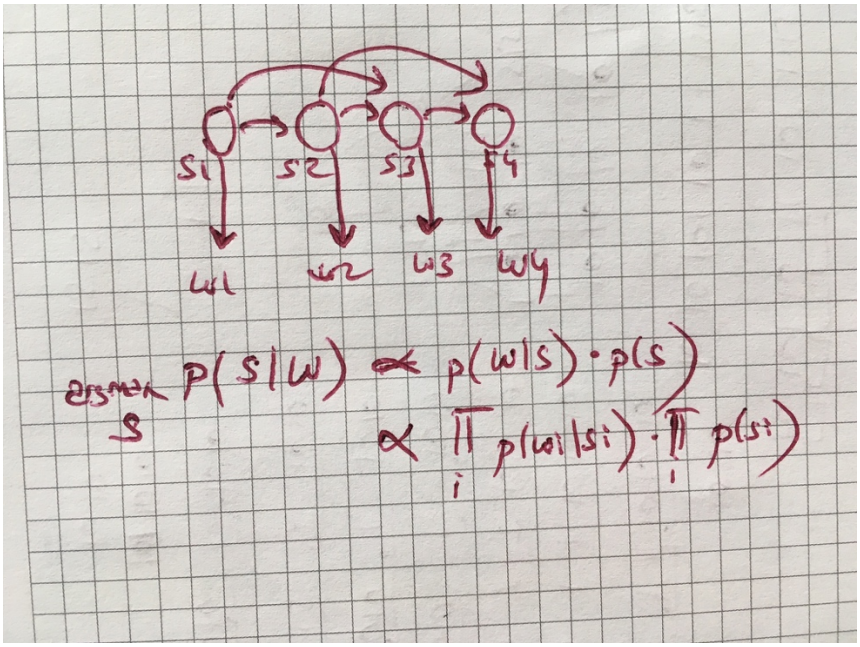
You should give this equation and define all the symbols and what they mean in this equation:

$$P(o|c) = \frac{\exp(u_o^T v_c)}{\sum_{w \in V} \exp(u_w^T v_c)}$$

u_o is the vector of the context word.
v_c is the vector of the centre word.
The numerator is the cosine similarity between these two vectors.
The denominator is the normalization which is computed for all possible contextual words for the given centre word c.

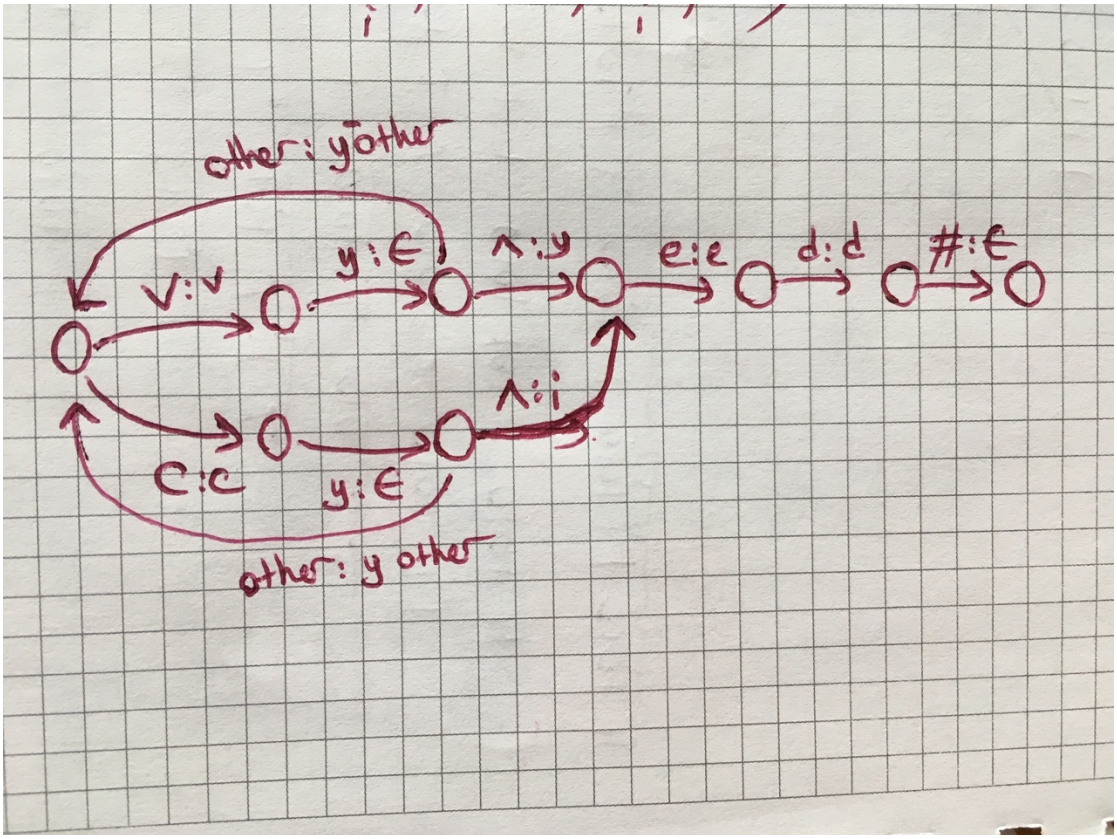
Part II. Design (each 10 points)

a) Design a trigram PoS tagging model using Hidden Markov Models. How will it look like? Draw properly. How will we predict the best tag sequence in that tagger? Describe mathematically with your equations.



b) Draw a finite state transducer (FST) for the English rule that drops 'y' at the end of a verb when it takes a past tense suffix. For example, 'copy+ed' becomes 'copied', 'study+ed' becomes 'studied', 'cry+ed' becomes 'cried', etc. However, 'obey+ed' remains 'obeyed', 'play+ed' remains 'played' etc., because of the vowel before the 'y' letter. Exceptions will not be handled. Only neat and readable drawings will be graded.

C: consonant
V: vowel
^: morpheme boundary
#: word boundary



c) You will build a text classification system. A number of articles on various topics are given. You will classify them into following categories: biology, geography, architecture, etc.

- What are the pre-processing tasks you will perform before training your model? Give your reasons. Are you going to use any tool/library for any of those tasks?

- Describe your machine learning model. You can define it either mathematically, or by drawing the architecture of your model along with a brief description of your equation or the architecture. Only the methods learned in the class will be graded.

We can apply stemming, normalization, spelling correction, removing punctuation, removing stopwords etc.
You can use Porter stemmer, NLTK library for other preprocessing tasks.

Preprocessing is 3 points.

You can apply Naive Bayes, neural networks by using word embeddings etc.

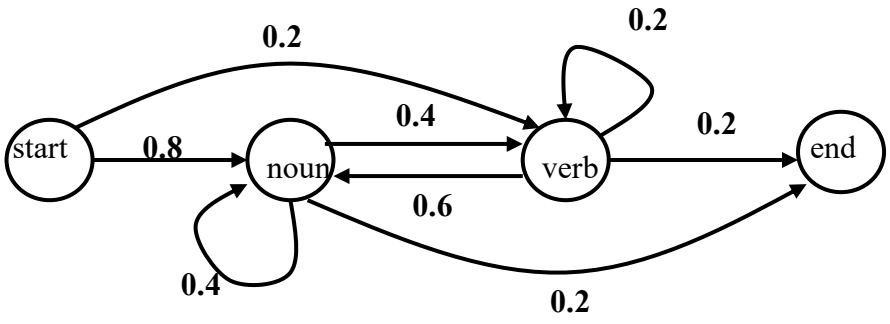
If your suggestion is Naive Bayes, you should give the equation for the Naive Bayes by defining your symbols. If your suggestion is using a neural network, then you should draw the architecture of your model. You should describe the inputs and the outputs of the model properly.

Part III: Problem Solving

a) (8 pts) We have a two word language: "fish" and "saw". Suppose in our training corpus,

- "fish" appears 4 times as a noun and 2 times as a verb.
- "saw" appears 6 times as a noun and 8 times as a verb.

Decide which word belongs to which part-of-speech tag according to the given hidden Markov model by applying Viterbi algorithm. Show your work. Brute force solutions will <u>not</u> be accepted.



	start	fish	saw	end
Start	1			
Noun	0	0.8*0.4	0.32*0.4*0.6 0.04*0.6*0.6	
Verb	0	0.2*0.2	0.32*0.4*0.8 0.04*0.2*0.8	0.02
End	0			

The answer is:
Start Noun Verb End

b) (9 pts) Give regular expressions for the languages generated by the following context-free grammars for a two-letter alphabet: {0, 1}:

a) S -> S1S1 | 0

0[01]*

Always begins with 10. The rest is irregular.

b) S -> SSSSS | 1 | 0

[10]([10]^4)*

These are not the only correct answers. All correct or partially correct answers were graded.

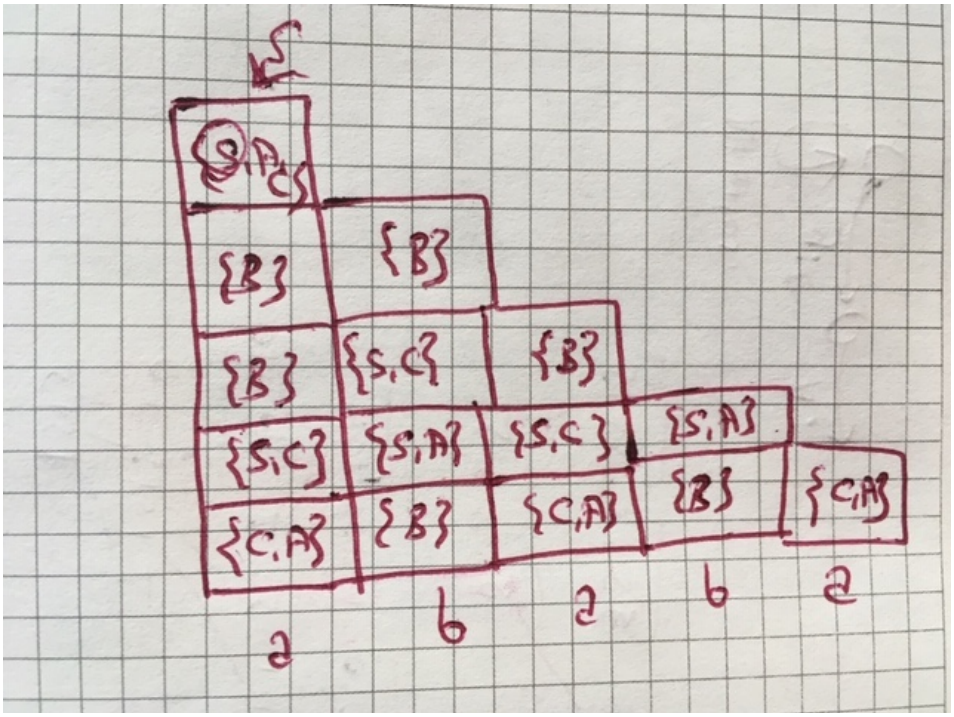
c) (8 pts) Apply CYK algorithm for the following example with the given grammar:

CNF grammar G (with capital letters are non-terminals and lowercase letters are terminals):

- S -> AB | BC
- A -> BA | a
- B -> CC | b
- C -> AB | a

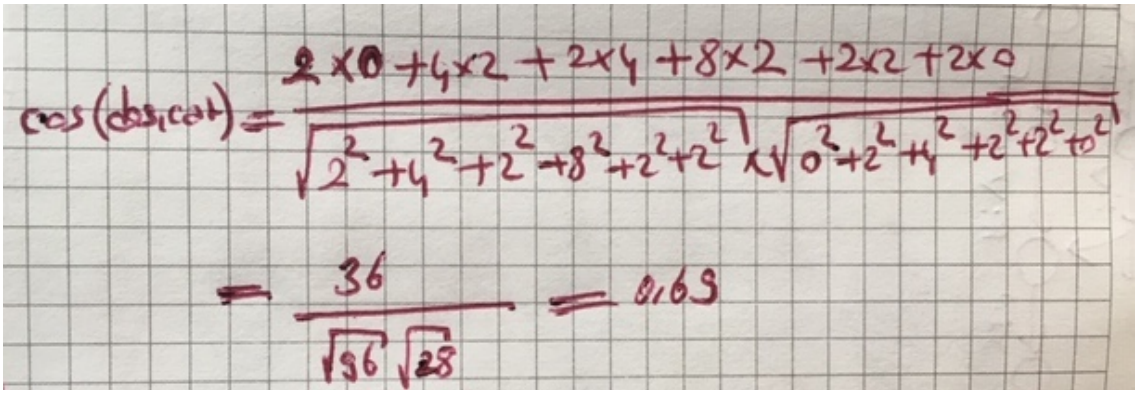
Is ababa in L(G), where L refers to a language with the specified grammar G?

Not just the answer but all of your scribbles should be provided. So I need to see that you solved it but not used any automatic tool on internet. If you just fill in the table, you get 4 points.



d) (4 pts) According to the given feature vectors for the target words *dog*, *cat*, *car*, and *bark*, calculate the similarity between 'dog' and 'cat' by using cosine similarity. Are these two words semantically similar to each other according to your solution? Explain briefly. (You can do approximations in your calculation.)

	Leash	**Walk**	**Run**	**Owner**	**Leg**	**Bark**
Dog	2	4	2	8	2	2
Cat	0	2	4	2	2	0

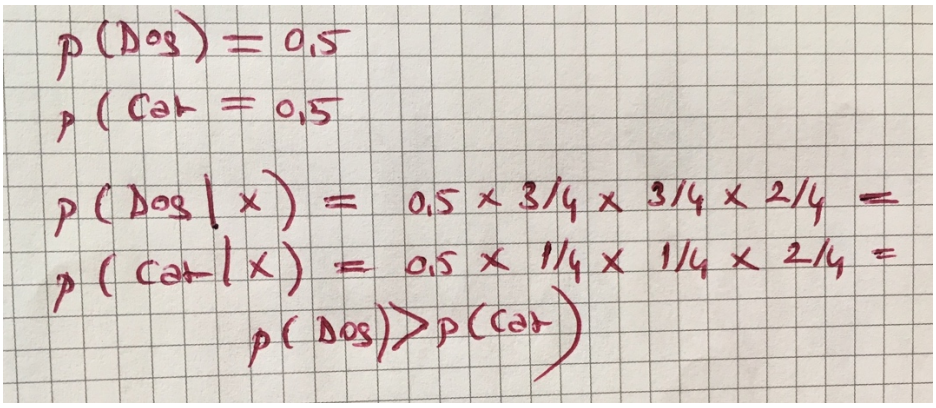


So two words are semantically similar to each other.

e) (7 pts) You are a robot in an animal shelter, and must learn to discriminate Dogs from Cats. You choose to learn Naïve Bayes classifier. You are given the following examples:

Example	**Sound**	**Fur**	**Color**	**Class**
Example #1	Meow	Coarse	Brown	Dog
Example #2	Bark	Fine	Brown	Dog
Example #3	Bark	Coarse	Black	Dog
Example #4	Bark	Coarse	Black	Dog
Example #5	Meow	Fine	Brown	Cat
Example #6	Meow	Coarse	Black	Cat
Example #7	Bark	Fine	Black	Cat
Example #8	Meow	Fine	Brown	Cat

Consider a new example, (Sound=Bark, Fur=Coarse, Color=Brown). Is it a dog or a cat? Apply Naive Bayes algorithm.

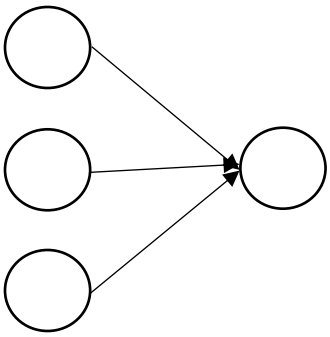


f) (10 pts) A neural network with three input neurons, and 1 output neuron is given below. Suppose that the output neuron uses an identity activation function, i.e., $z = f(z)$ where z is the total input to the neuron. Let d be the desired output and d=4, and let $E = -d\,logz - (1 - d)\log(1 - z)$ be the cross entropy error. For the inputs (1,2,1) and weights (1,3,1), perform a forward pass to compute the output of the network and then apply backpropagation with gradient descent for one iteration. Write down the weights after the forward pass and also after the backward pass separately. Learning rate $\mu = 10$.

Hint: The derivative of the identity function is simply 1. The derivative of logarithm is given below:

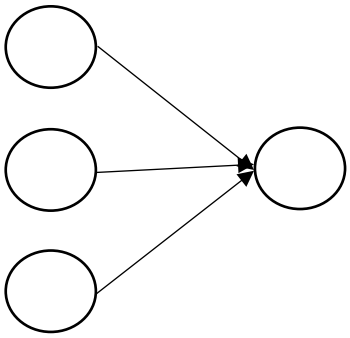
$$\frac{\partial logu}{\partial u} = \frac{1}{u} \qquad \frac{\partial \log(1-u)}{\partial u} = \frac{-1}{1-u}$$

After forward pass:



1*1+2*3+1*1 = 8

After backward pass (backpropagation):



$$\frac{dE}{dz} = -d\frac{1}{z} - \frac{d-1}{1-z}$$

$$= \frac{-4}{8} - \frac{3}{-7} = -0,071$$

$$\Delta w_{z1} = 10 \times (-0,071) \times 1 = -0,7$$

$$\Delta w_{z2} = 10 \times (-0,071) \times 2 = -1,4$$

$$\Delta w_{z3} = 10 \times (-0,071) \times 1 = -0,7$$