NTNU – Trondheim
Norwegian University of
Science and Technology

Department of Computer Science (IDI)

Examination paper for TDT4310: Intelligent Text Analytics and Language Understanding

Academic contact during examination:	**Prof. Björn Gambäck**
Phone:	**+46 70 568 15 35**
Examination date:	**May 22nd, 2019**
Examination time (from-to):	**09:00 – 13:00**
Permitted examination support material:	**D (no books or notes allowed)**

Other information: Since all the course material and the lectures were in English, the exam is too. However, answers may of course be given in either English or Norwegian.

Read the questions carefully. If you consider that some information needed to solve a question is missing, give a short account of the assumptions you have found it necessary to make.

Maximum number of points on the exam: 75. The exam counts for 75% of the overall course grade. There are 15 questions, the first ten count for 4 points each, the remaining five for 5 points each. However, this does neither imply that the questions in each group are at an equal level of difficulty, nor that the answers to those questions should be of about equal length. The opposite is rather the case, so some questions are most certainly easier than others and require shorter answers. It will thus make sense to first reply to those. Hence, be aware that the questions in each group are not necessarily ordered by level of difficulty, so read all questions carefully before starting to answer the ones you consider “low-hanging fruit”.

Language:	**English**
Number of pages (front page excluded):	**4**
Number of pages enclosed:	**6**

Checked by:

Date Signature

Students will find the examination results in Studentweb. Please contact the department if you have questions about your results. The Examinations Office will not be able to answer this.

This page intentionally left blank.

TDT4310: Exam Questions 2019 [maximum points: 75]

4-point questions:

1) **What are the main features of human languages that make parsing of these different from the parsing of programming languages?**

2) **"Tokenization in English is straight-forward: each word is delimited by a space." Give three examples which suggest that this claim is oversimplified.**

3) **How would you go about dividing your data set in order to carry out a thorough evaluation?**

4) ***Alignment* is a key concept in training statistical machine translations systems from parallel text. Explain the term using examples?**

5) **What kind of constructions can be problematic for**
   a) **a hypothesis-driven parser (top-down)?**
   b) **a data-driven parser (bottom-up)?**

6) **Information retrieval systems can be used in different ways. Sometimes optimizing the system's performance as regards retrieval recall is crucial; sometimes the retrieval precision is more important. Give examples of both cases with brief motivations.**

7) **What is the significance of the terms "*subjectivity*" and "*polarity*" in textual sentiment analysis? Give examples and discuss the differences between subjective and objective statements.**

8) ***Grounding* is an important concept for dialogues. Describe the phenomenon and ways in which it can be achieved in the three basic types of dialogue systems, i.e., those where the dialogue management is system driven, user driven as well as built on mixed initiative.**

9) **Suppose someone took all the words in a sentence and reordered them randomly. What would you need to do if you were to write a program that would take as input such a *bag of words* and produce as output a guess at the original order?**

10) **In the context of semantic interpretation:**
   c) **Why can it sometimes be advantageous to *avoid* resolving all semantic ambiguities in a sentence?**
   d) **State the *Principle of Compositionality.***

5-point questions:

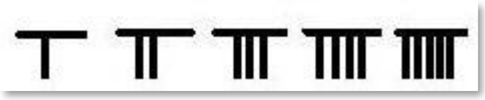
11) **Describe advantages and disadvantages with interlingua-based, transfer-based, direct and neural machine translation. Argue for why you think that one of them (or some other translation strategy!) is better than the other when translating technical manuals.**

12) **The company Predictor Ltd released program for word prediction based on trigrams.** The idea was that the program would suggest the next word based on the two previous words typed by the user. However, when evaluating the program, it soon became apparent that several plausible trigrams had zero probability. **Why did this happen? Why is it a problem? Suggest a way to solve the problem.**

13) The 11th century Irish text *Lebor Gabála Érenn* tells how king Fénius Farsaid created *Goídelc,* the Gaelic language, by taking the best pieces of the 72 languages that according to the Bible appeared after "the confusion of tongues" through the Tower of Babel.

a) Which language is the best is obviously subjective, but suppose that Fénius had been a Computational Linguist and **suggest three basic features that he should have included in the language** (i.e., properties that would make a language "good" from a Computational Linguistic perspective).

b) Fénius Farsaid went on to discover alphabets. First Hebrew, Greek and Latin, and finally Ogham (Old Irish) which supposedly is the most perfected – because it was the last. Suppose that you wanted to create a perfect writing system, but from a Computational Linguistic perspective. **Suggest at least one property that this writing system should have.**



B L F S N – the first 5 characters of Ogham (*beith-luis-nin*)

14) Given the following Context-Free Grammar, CFG (the numbers are not part of the rules):

1. S → NP VP	2. VP → VP PP	3. PP → P	4. IV → runs
5. NP → N	6. VP → VP CONJ VP	7. PP → P NP	8. C → that
9. NP → D N	10. N → squirrel	11. TV → chases	12. P → in
13. NP → NP CONJ NP	14. N → he	15. TV → eats	16. P → away
17. VP → IV	18. N → John	19. TV → catches	20. CONJ → and
21. VP → IV PP	22. N → Mary	23. TV → tells	24. D → the
25. VP → TV NP	26. N → dog	27. TV → sees	
28. VP → TV C S	29. N → tree	30. IV → sits	

Consider the following story:

(A) John sees the dog and Mary sees the dog. (B) The dog sees John and Mary.
(C) The dog sees a squirrel. (D) The squirrel sits in a tree. (E) That squirrel sees the dog.
(F) The squirrel was seen by the dog. (G) The dog runs. (H) The squirrel in the tree runs.
(I) The dog chases the squirrel and eats the squirrel. (J) The dog eats.
(K) John sees the dog eat the squirrel. (L) John tells Mary that the dog eats the squirrel.
(M) The dog sees that John sees that he eats the squirrel. (N) And the dog runs away.
(O) Mary and John chase the dog. (P) John chases and catches the dog. (Q) John eats dog.

a) Several of the sentences in the story are not well-formed according to the CFG. **List the sentences that the CFG can generate** (ignore the periods).

b) Not all sentences that the CFG can generate are actual English sentences. **Give three examples of sentences generated by the CFG that would be considered ungrammatical in standard English.**

c) **One of the rules in the CFG is redundant: any sentence that can be generated using this rule can already be generated by a combination of other rules. What is the number of that rule?**

15) Your spaceship has just crashed and you try to read the manual for spaceship repair when a friendly-looking being from the planet Rigel sidles up to you and says:

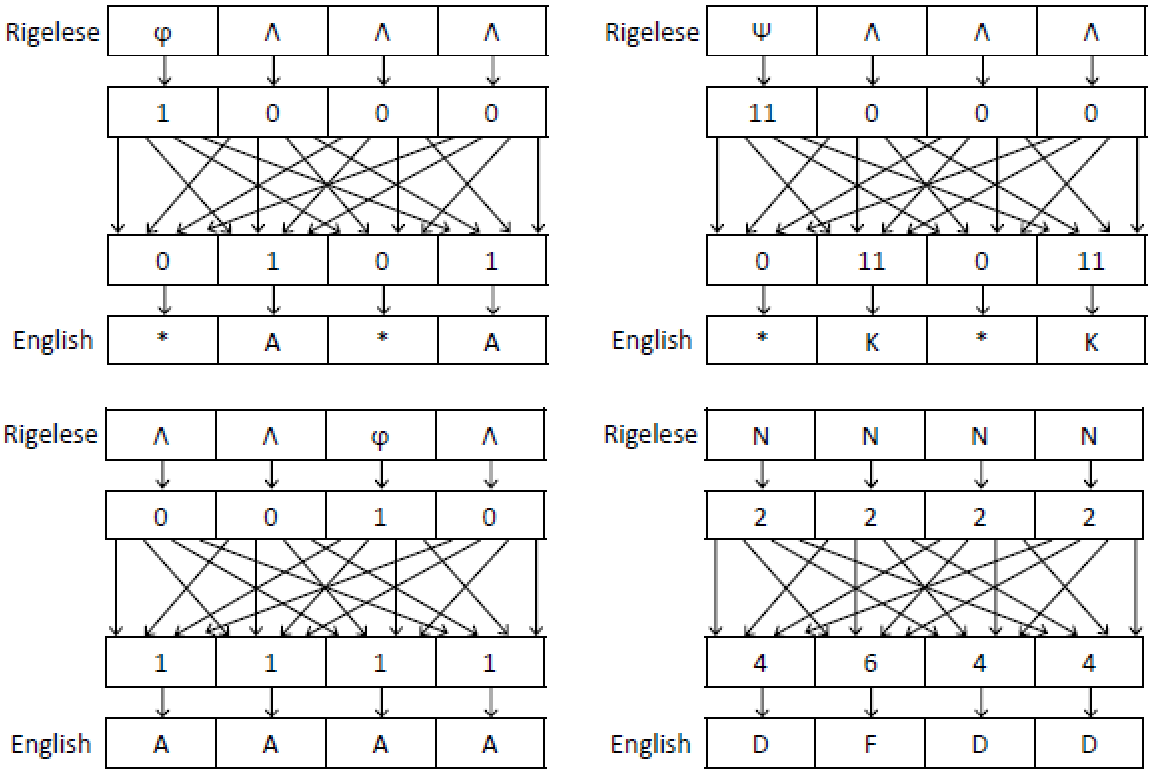
> “ζΨδξ ϖΞΝϕ ΑϕΩυ ΠΠαΣ”

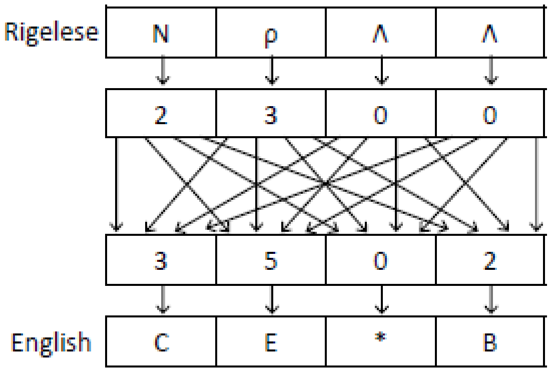
Luckily for you, you own the GalactiLang™ translation device, which can translate from the Rigelese sound system into English. The translator first turns a Rigalese word into four numbers, then uses a neural network to transform those numbers into another sequence of four numbers, that finally are transformed into English letters using the following table:

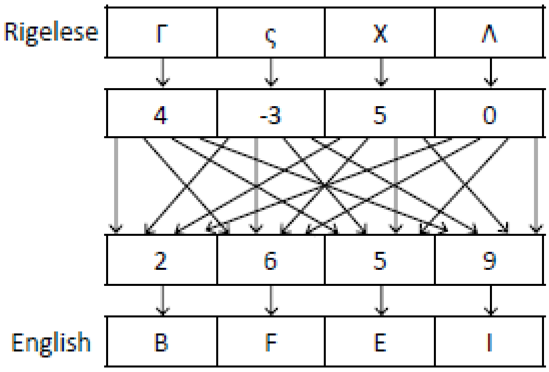
0	1	2	3	4	5	6	7	8	9	10	11	12	13
*	A	B	C	D	E	F	G	H	I	J	K	L	M

14	15	16	17	18	19	20	21	22	23	24	25	26
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Here are six examples of the translator in action:



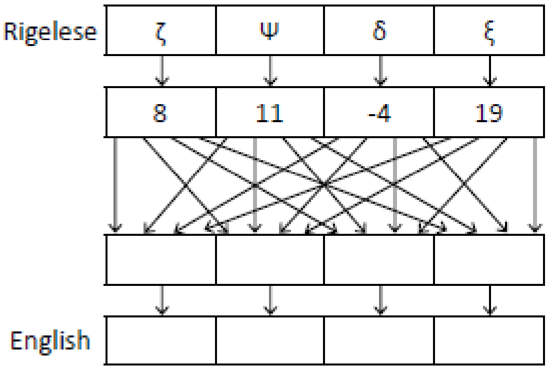




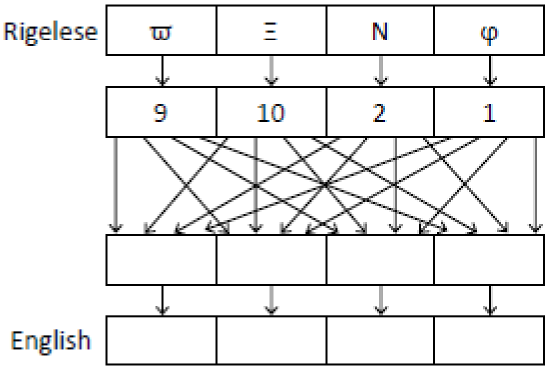
As you can see, the network in fully connected, with the value of a new cell being determined by the sum of all its input times the weight on each input arrow.

Unfortunately, when you try to translate the message from the Rigelian, your translator runs out of power after only computing one step of the translation. As a result, this is all that it gives you:

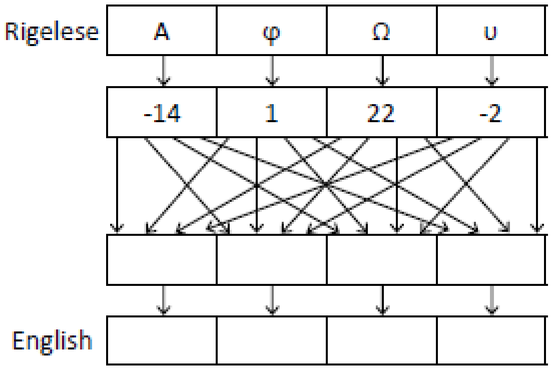
Word 1



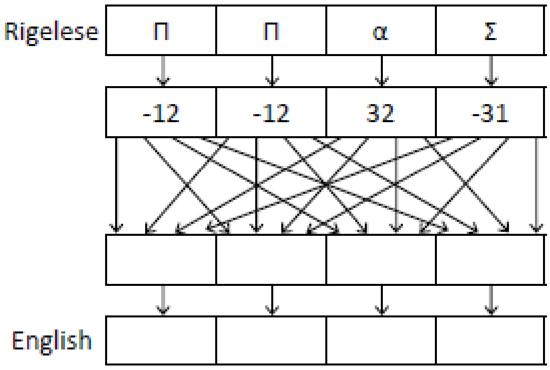
Word 2



Word 3



Word 4



Finish the (four word) translation that the translator started. Although you can see the six example translations above, you do not know what weights are attached to each arrow in the diagram (although you do know that the weights are the same across the translation of all four words). Therefore, you will have to use those diagrams to figure out the inner workings of the translator.