TDT4310: Exam Questions & Answers 2019-05-22

4-point questions:

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

Human languages are flexible and evolving [1p]. They contain ambiguity and redundancy [2p]. The human mind's memory capacity is limited (which is reflected in the possible complexity of the human languages) [1p].

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

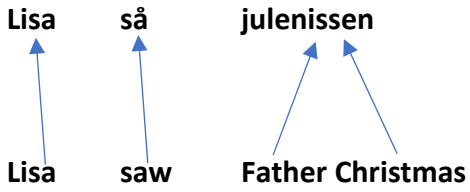
Abbreviations [1p]. Contractions [1p]. Social media language (hashtags, emoticons, etc.) [2p]. Compounds, hyphenation and/or multi-word tokens [2p]. Numbers and other symbols [1p].

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

Training / development (validation) / (unseen) test set [4p]
and/or use *k*-fold cross-validation [4p]: Break up data into *k* folds (equal positive and negative inside each fold?). Choose one fold as a temporary test set; train on the other folds; average performance over *k* runs.

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

An alignment is a mapping [2p] from a TL word to one SL word (or NULL) [1p], e.g., [1p]:



5) What kind of constructions can be problematic for

a) a hypothesis-driven parser (top-down)?

Left recursive structures [2p].

b) a data-driven parser (bottom-up)?

Empty production rules [2p].

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.

Optimizing recall is important if we need high coverage (quantity, e.g., searching for patents or laws) [2p].
Precision is more important if we need high relevance (quality, e.g., in a web search) [2p].

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.

Subjectivity = objective (factual) or subjective (has sentiment / is opinionated) [2p].
Polarity (if subjective) = positive or negative [2p].

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.

Common ground = set of things mutually believed by both speaker and hearer. The hearer must *ground* or acknowledge the speaker's utterances [2p.]
(*Principle of closure* **=** Agents require evidence that they have succeeded in performing an action [1p].)
System-driven / User driven / Mixed initiative: relevant contribution, paraphrasing, repeating, explicit acknowledgement, ask for clarification / follow-up questions, discourse markers / continued attention [2p].

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?

If you just randomly guess the original word order among *n* words, there are of course *n!* possible combinations – and the enormous search space thus would give only a very minute chance of guessing the original word order.
Smarter: build a language model, e.g., by collecting a large corpus [1p] and collecting n-gram statistics, which can then be utilised through the Markov assumption for the sequence prediction (i.e., assuming dependence on only a restricted number of previous words) and/or by using a noisy channel model. The perplexity can be further reduced, e.g., by part-of-speech tagging your bag-of-words and the corpus, or by using a shallow parser. Additionally (or), a machine learner can be trained on the corpus to predict the most plausible sequences. [3p]

10) In the context of semantic interpretation:

c) Why can it sometimes be advantageous to *avoid* resolving all semantic ambiguities in a sentence?

Some ambiguity might best be left to the users to resolve and/or be translated in the same (ambiguous) way in a machine translation system. Some ambiguity may be intentional and should be kept (e.g., in a joke or a poem).

d) State the *Principle of Compositionality.*

Meaning ultimately flows from the lexicon. Meanings are combined by syntactic information.
The meaning of the whole is a function of the meaning of its parts (= the substructure given by syntax). [2p]

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.

Interlingua, pros: can easily cover many languages; cons: knowledge representation [1p].
Transfer, pros: can give good quality; cons: difficult to cover many languages / rule-writing [1p].
Direct, pros: easy to implement; cons: difficult to cover many languages / lexica, possibly low quality [1p].
Neural, pros: can be fast to build; cons: requires a lot of data, unpredictable output/quality [1p].
Decent argumentation for either alternative = 1p (although transfer probably is the best option in this case).

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.**

Sparse data: 0 probability for unseen trigrams [2p].
Solve using smoothing and/or back-off to uni-/bigrams and/or linear interpolation [3p].

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 <u>three</u> basic features that he should have included in the language** (i.e., properties that would make a language "good" from a Computational Linguistic perspective).

Restrictive grammar. Simple and regular morphology. Reduced ambiguity and redundancy. [3p]

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 <u>one</u> property that this writing system should have.**

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

Easy to encode (e.g., simple segmentation and tokenization). One-to-one phoneme-morpheme mapping. [2p]

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 <u>can</u> generate** (ignore the periods).

The CFG can generate sentences B, G, I, M and Q [2p].

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.**

Many possibilities, e.g., “squirrel chases”, “the he sits dog”, “Mary and John runs” [1p].

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?

Redundant rule: 21 (VP → IV PP) [2p]. It is already covered by combining rules 17 (VP → IV) and 2 (VP → VP PP).

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:

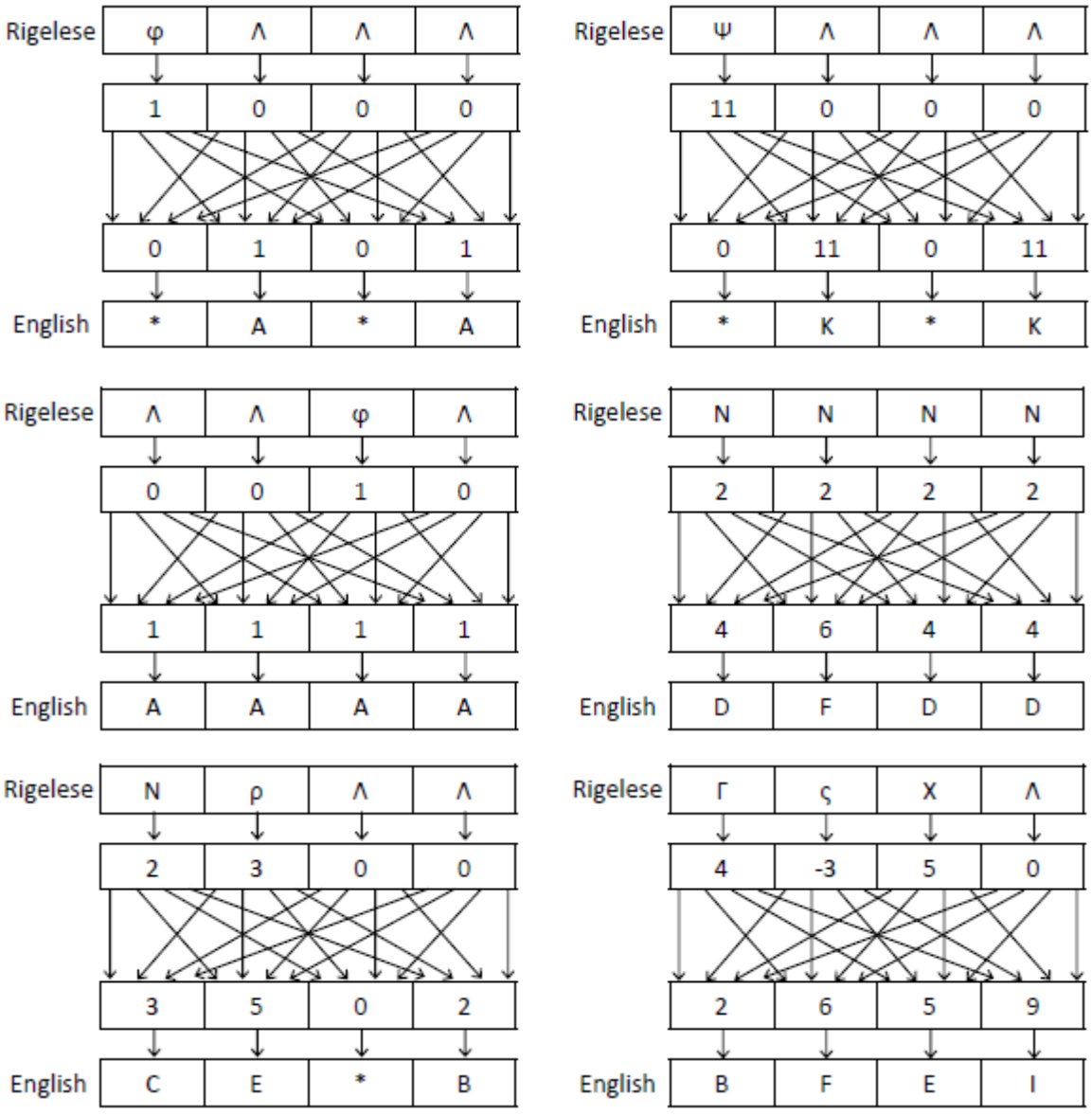
"ζ Ψ δ ξ ϖ Ξ N ϕ A ϕ Ω υ Π Π α Σ"

Luckily for you, you own the GalactiLang™ translation device, which can translate from the Rigelese sound system into English. The translator first turns a Rigelese 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

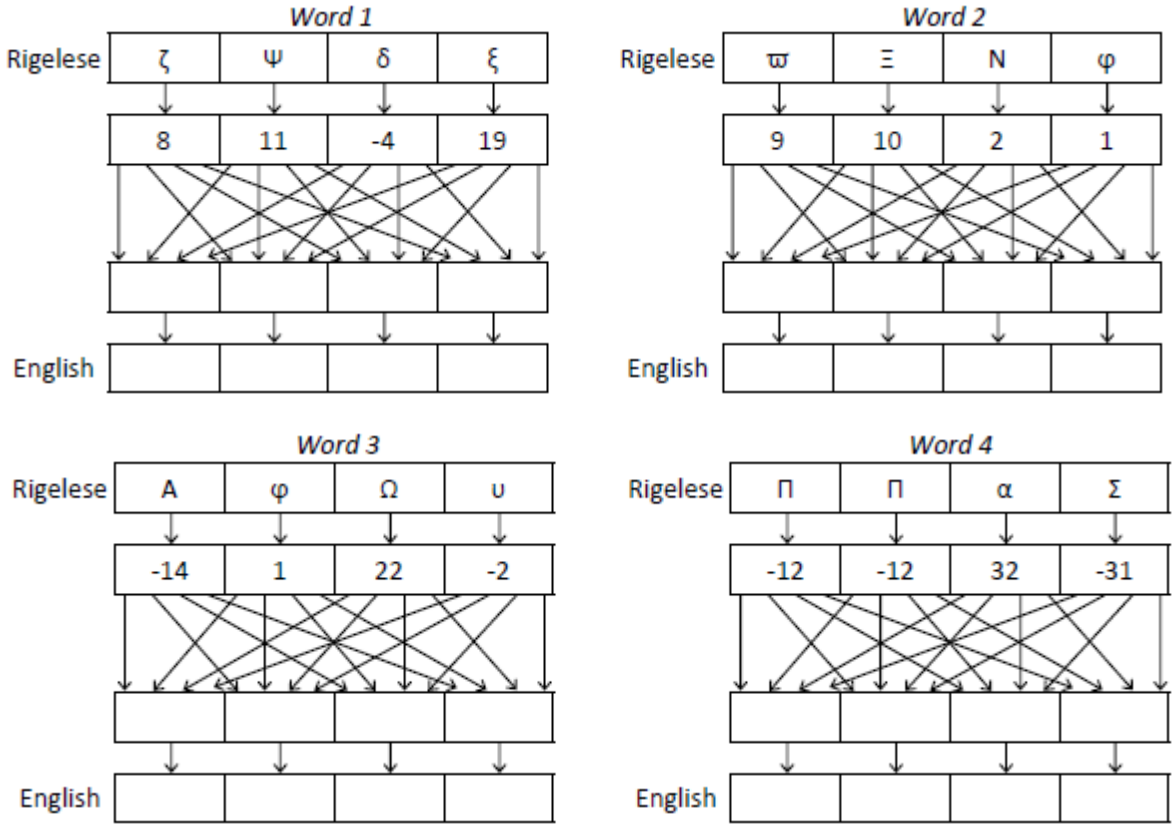
14	15	16	17	18	19	20	21	22	23	24	25	26
N	O	P	Q	R	S	T	U	V	W	X	Y	Z

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:



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.

GOOD LUCK WITH THAT [5p].

We can formulate the Rigelese (R) to English (E) translation as E=WR, with the weight matrix W:

$$W = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

Or we can simply observe from the six examples that if we call the four input elements < a , b , c , d >, then the four outputs will be (from left to right): (b + c), (a + b + c), (c + d) and (a + c).