IMPERIAL COLLEGE LONDON

TIMED REMOTE ASSESSMENTS 2021-2022

MEng Honours Degree in Mathematics and Computer Science Part IV
MEng Honours Degrees in Computing Part IV
MSc Advanced Computing
MSc Artificial Intelligence
MSc in Computing (Specialism)
for Internal Students of the Imperial College of Science, Technology and Medicine

*This paper is also taken for the relevant assessments for the*
*Associateship of the City and Guilds of London Institute*

# PAPER COMP70016=COMP97115=COMP97116

# NATURAL LANGUAGE PROCESSING

Thursday 24 March 2022, 10:00
Writing time: 120 minutes
Upload time: 30 minutes

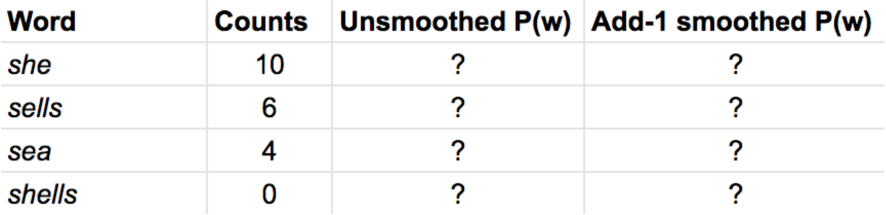
*Answer ALL THREE questions*
Open book assessment

This time-limited remote assessment has been designed to be open book. You may use resources which have been identified by the examiner to complete the assessment and are included in the instructions for the examination. You must not use any additional resources when completing this assessment.

The use of the work of another student, past or present, constitutes plagiarism. Giving your work to another student to use constitutes an offence. Collusion is a form of plagiarism and will be treated in a similar manner. This is an individual assessment and thus should be completed solely by you. The College will investigate all instances where an examination or assessment offence is reported or suspected, using plagiarism software, vivas and other tools, and apply appropriate penalties to students. In all examinations we will analyse exam performance against previous performance and against data from previous years and use an evidence-based approach to maintain a fair and robust examination. As with all exams, the best strategy is to read the question carefully and answer as fully as possible, taking account of the time and number of marks available.

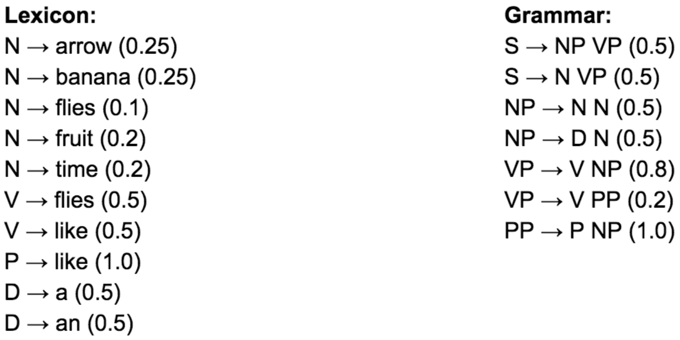
Paper contains 3 questions

1 a Complete the following table of a unigram language model where P(w) stands for the probability of a word/token w, and all tokens in the language are shown in the table.



| Word | Counts | Unsmoothed P(w) | Add-1 smoothed P(w) |
|---|---|---|---|
| *she* | 10 | ? | ? |
| *sells* | 6 | ? | ? |
| *sea* | 4 | ? | ? |
| *shells* | 0 | ? | ? |

b Given the following probabilistic grammar and lexicon:



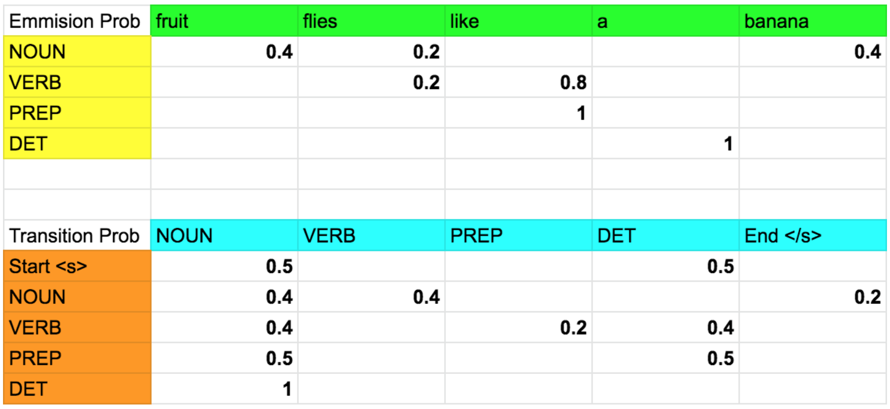
**Lexicon:**
N → arrow (0.25)
N → banana (0.25)
N → flies (0.1)
N → fruit (0.2)
N → time (0.2)
V → flies (0.5)
V → like (0.5)
P → like (1.0)
D → a (0.5)
D → an (0.5)

**Grammar:**
S → NP VP (0.5)
S → N VP (0.5)
NP → N N (0.5)
NP → D N (0.5)
VP → V NP (0.8)
VP → V PP (0.2)
PP → P NP (1.0)

Provide the CKY parse matrix for the following sentence:

`flies like a banana`

c Consider the following Hidden-Markov Model (HMM) for Part-of-Speech (PoS) tagging:



| Emmision Prob | fruit | flies | like | a | banana |
|---|---|---|---|---|---|
| NOUN | 0.4 | 0.2 | | | 0.4 |
| VERB | | 0.2 | 0.8 | | |
| PREP | | | 1 | | |
| DET | | | | 1 | |

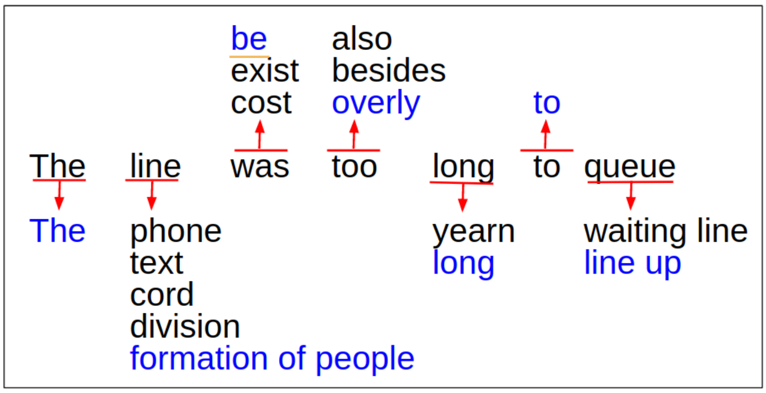
| Transition Prob | NOUN | VERB | PREP | DET | End </s> |
|---|---|---|---|---|---|
| Start <s> | 0.5 | | | 0.5 | |
| NOUN | 0.4 | 0.4 | | | 0.2 |
| VERB | 0.4 | | 0.2 | 0.4 | |
| PREP | 0.5 | | | 0.5 | |
| DET | 1 | | | | |

Apply the Viterbi algorithm to PoS tag the following sentence. As part of your answer, draw the matrix indicating the probabilities for each state. Give the final tags for each word: `fruit flies like a banana .`

d Explain two advantages and one disadvantage of learning a character-level language model compared to learning a token-level counterpart.

*The four parts carry, respectively, 20%, 30%, 20%, and 30% of the marks.*

2 a In the problem of all-words sense disambiguation, given a sentence, the task is to select one among the possible senses (meanings) of each word. Some words are ambiguous and therefore have multiple possible sense labels, while others are not ambiguous and therefore have only one sense label, for example:



Consider the following settings: In a corpus of 100,000 sentences, each sentence is annotated with a sense label for each of its words. The input vocabulary size is 10,000 distinct words. 90% of these words have 2-7 different meanings (4 meanings on average), while 10% of the words have only one meaning (e.g. ‘the’ and ‘to’ above). Some senses are very rare, whereas others are very frequent. For example, out of 100 sentences with `line` as the ambiguous word, 90 have `cord` as the sense.

i) Assume you are using a Recurrent Neural Network (RNN) to address this problem. Would this be a many-to-one or many-to-many RNN? Explain why.

ii) What will the possible labels be? Approximately, what is the maximum number of distinct labels?

iii) Provide a metric that would be appropriate to report average performance of the model on all words. Justify your answer.

b Consider the task of hate speech detection in social media posts for 50 languages, all using Latin alphabet, from high-resource, such as English, to low-resource, such as Romanian. Given a post, your multilingual model has to categorise it as `hateful` or `not-hateful`. Assume you have labelled training data for all languages, but in different sizes: from thousands of instances for English to only a few dozen or a few hundreds for many low-resource languages. Many of these languages are not in any pre-trained representation (like BERT), so pre-trained embeddings cannot be used.

i) Would you use BPE for this problem? Why?

ii) Consider building a CNN classifier to address this problem. Assume no BPE. Would you use a token-level or character-level CNN? Explain why.

c Transformers and (attention-based) RNNs are two architectures commonly used in encoder/decoder tasks. Give one conceptual difference between the Transformer encoder and the RNN encoder, and one conceptual difference between the Transformer decoder and the RNN decoder.

*The three parts carry, respectively, 55%, 25%, and 20% of the marks.*

3 a Consider a single-head self-attention computation in a Transformer encoder. Using the Equation (1) below and the following variables:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

$X = [[3, 2, 1], [0, 1, 1]]$ (Input at the current layer)
$Wq = [[1, 0, 0], [0, 1, 0], [0, 0, 1]]$ (Query projection matrix)
$Wk = [[0, 1, 0], [0, 0, 1], [1, 0, 0]]$ (Key projection matrix)
$Wv = [[0, 0, 1], [1, 0, 0], [0, 1, 0]]$ (Value projection matrix)

i) Compute the attention matrix $QK^T$, indicating your computations in the answer.

ii) Calculate the output from the attention computation described in Eq. (1) above, indicating your computations in the answer.

b Consider a standard BERT model.

i) Part of the BERT model's objective is Masked Language Modelling (MLM). In MLM, 15% of the words in an input sequence are randomly replaced with a [MASK] token. These [MASK] tokens are additionally processed by the following strategy:

M1. 80% of the time they're kept as a [MASK] token

M2. 10% of the time they're replaced by a random token from the vocabulary

M3. 10% of the time they're replaced by the original masked word.

What is a reason behind masking strategies M2 and M3?

ii) The task of named entity recognition (NER) is defined as seeking "to locate and classify named entities mentioned in unstructured text into predefined categories such as person names, organisations and locations".

Assume we have 4 labels: `N[ame]`, `O[rganization]`, `L[ocation]` and `E[lse]`, where `E` is the label for tokens which are not named entities.

Here's an example of a sentence and its labels:

```
[Bill, Gates, founded, the, company, Microsoft, with,
headquarters, in, Seattle]
[N, N, E, E, E, O, E, E, E, L]
```

Describe how you would use BERT to perform NER using the above 4 possible labels. As part of your answer, explain the type of classification problem this is, the loss function you would use, and the inputs and outputs shapes.

c The following questions are related to the number of parameters in the Transformer architecture.

   i) As we increase the dimensionality $d$ of the input/model, specify how the total number of parameters in the model will change with reference to $d$ for the: (1) self-attention layer, and (2) position-wise feed-forward layer?

   ii) Does the number of parameters in the self attention and position-wise feed-forward layers depend on the input sequence length? Why?

   iii) A shortcoming of the Transformer model is that they become computationally expensive to use on long sequences because each token attends to every token in the sequence. Its complexity is $O(n^2)$, i.e. quadratic with respect to input length. A proposed solution is a sliding-window self-attention mechanism, whereby each token only attends to a fixed window size $k$ of tokens either side of itself. Describe what impact this proposed solution has on the computational complexity of the self-attention layer. Make reference to the original complexity.

*The three parts carry, respectively, 20%, 30%, and 50% of the marks.*