Coursework commentary 2018–2019

CO3354 Introduction to natural language processing

Coursework assignment 2

General remarks

Students were reminded that:

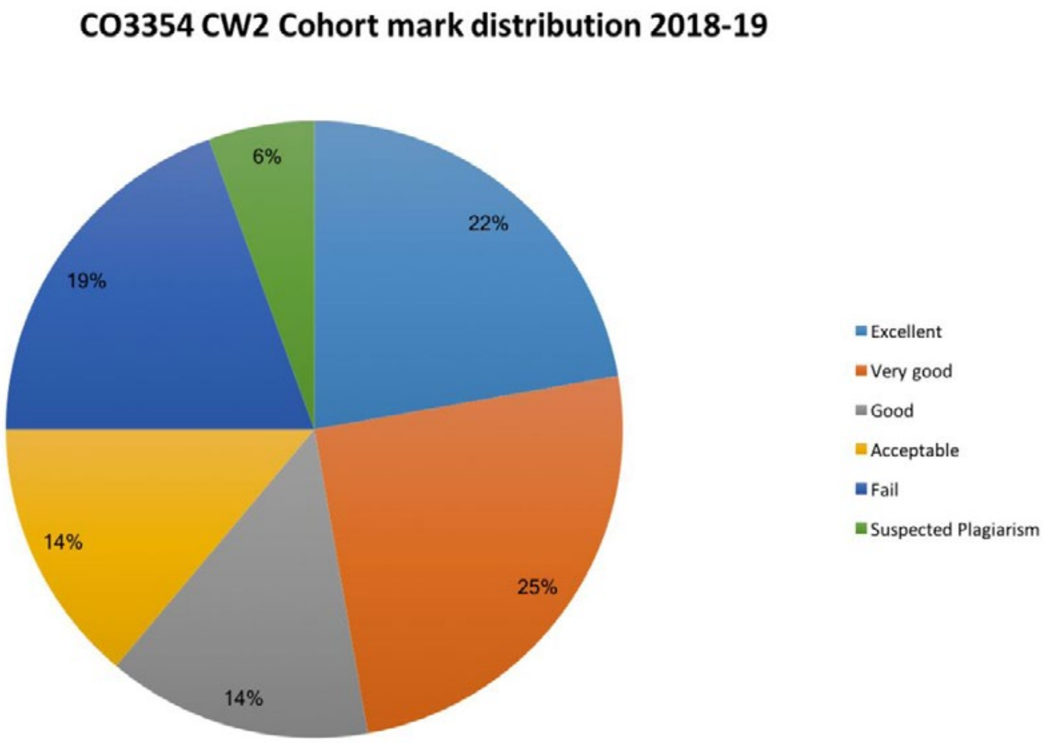
> *It is important that your submitted coursework assignment is your own individual work and, for the most part, written in your own words. You must provide appropriate in-text citation for both paraphrase and quotation, with a detailed reference section at the end of your assignment (this should not be included in any word count). Copying, plagiarism and unaccredited and wholesale reproduction of material from books or from any online source is unacceptable, and will be penalised (see our guide on* how to avoid plagiarism *on the VLE).*

Students were also advised to:

- explain answers and show working (where applicable) for full marks. Main results should be given in the body of the answer, rather than relegated to appendices or additional files.
- submit work as a single PDF file; to include Appendices with any Python code that was written, and the results of running the code. Students who used Jupyter notebooks (recommended) could additionally download the notebook in .ipynb format and submit it as a separate file.
- make sure code is adequately commented – this can be done using the Markdown option in Jupyter. Comments should be grammatical, concise and avoid stating the obvious.

Some students lost marks unnecessarily by disregarding one or more of these points.

See the 2018–2019 CW2 cohort mark distribution below:



Comments on specific questions

Question 1: Part-of-speech tagging

This question involved the use of techniques and corpus resources described in the NLTK book and the subject guide for marking up NL texts with grammatical categories. It was designed to reinforce students' knowledge and understanding of these categories and resources as well as the use of regular expressions.

a. Students gained marks for successfully locating the 'treebank' corpus, running the evaluation and devising and testing linguistically motivated extensions to the RE tagger. Given that the question specifically mentions 'closed-class words such as prepositions and conjunctions', it was surprising that not all students included them. When working with REs it is important to remember when to use the start and end of string markers '^' and '$'; there is a risk of generating spurious results, e.g. the string '*ing' without a closing '$' would match words like 'finger' as well as gerund forms of verbs like 'walking'. The majority of students scored very good or excellent marks for this sub-question.

b. The previous question involved hand-coding patterns for tagging; this is an instructive exercise, but better results are obtained by training taggers using machine learning. This sub-question involved the backoff technique, applying different types of taggers in sequence. Generally, students showed good understanding of this process and most got very good or excellent marks. However, some common errors included:

   - chaining directly from the bigram to the default or RE tagger, rather than via the unigram tagger
   - using the tag from the most common word in the corpus for the default tagger, rather than the most common tag
   - training and testing on the same dataset.

   Very few students gave good answers to (iii). Good answers would have started by looking at instances of '-NONE-' in context; many made vacuous statements to the effect that items are tagged '-NONE-' because they are not part of the language. A small number did some research and found that this tag is actually used for annotating ‘traces’ or ‘empty nodes’ to indicate coreference and such, which is an artefact of the particular grammatical formalism that underlies the tagset.

c. This involved applying techniques from the previous sub-question to naturally-occurring text, rather than NLTK’s pre-formatted corpus resources. Good answers to (i) started by segmenting the full text into sentences and then tokenising each sentence as a list of words: some answers simply treated the entire text as a list of words, losing sentence boundary information which could compromise the accuracy of the tagger. Good answers would also have cleaned up the file by stripping out front and back matter, which is not part of the original text. Apart from these complications, this was a fairly straightforward exercise and most students obtained very good or excellent marks.

d. This sub-question was intended to encourage students to read beyond the prescribed course materials. Any reasonable, well informed and supported answers were accepted; around half the students were awarded good, very good or excellent marks, though a sizeable minority simply did not attempt to answer.

Question 2: Information extraction

a. Part (a) was **bookwork** which students should have been able to tackle successfully through carefully reading the subject guide and recommended sources. With questions like this, it is important to remember to answer **in your own words** and to avoid defining technical expressions in terms of other (unexplained) technical terms. Students should also avoid circular/repetitive answers such as ‘*Named Entity Recognition refers to automatically recognizing entities*’, ‘*Relation extraction refers to identifying semantic relations*’, “’ *process [NER] is difficult due to the difficulty of accurately identifying named entities within text*’, etc.

b. Part (b) required students to undertake independent reading and report on their findings. Marks were awarded for:

- evidence of technical knowledge/understanding
- structure, coherence and clarity of answers
- scholarly practice.

Generally, students did slightly less well here than on the more problem-solving oriented questions. Some answers were thoroughly researched and well presented, but others were rather brief, lacked detail, failed to reference specific examples or did not properly address the question. Just over half the answers to part (b) were marked good, very good or excellent, but fewer than half the answers to part (c) reached this standard.

c. See part (b) above.