Coursework commentary 2017–2018

CO3354 Introduction to natural language processing

Coursework assignment 1

Learning outcomes

This assignment addressed learning outcomes 1 and 2 from the subject guide:

1. utilise and explain the function of software tools such as corpus readers, stemmers, taggers and parsers
2. explain the difference between regular and context-free grammars and define formal grammars for fragments of a natural language.

General remarks

- You should list all references at the end of your work and they should be properly cited whenever referred to. Answers that consist largely of quoted material are unlikely to get high marks, even if properly referenced.
- You should **explain** your answers and show working (where applicable) for full marks. Answers lacking motivation risk getting low marks, even if they give correct results.
- You are required to submit your work as a PDF file, with an appendix including any Python code you have written and the results of running your code. Marks may be deducted if you do not submit your work in the required format.
- Some students made things hard for themselves by coding functions from scratch which are provided as built-in functions in NLTK, e.g. tabulate, collocations. They did not normally get extra marks for doing so.
- Your main results should be given in the body of the answer. It is time-consuming for markers to be referred to the appendix or a separate file and to have to sift through pages of code.

Comments on specific questions

Question 1

Syntax and formal grammars

a. i. This question or something very like it comes up frequently in coursework and/or exams, but many students failed to show that they have grasped the distinction.

Formally: both CFGs and RGs consist of sets of production rules with a single symbol on the LHS, of the general form X →Y …

Good answers would compare the two types of grammar under the following headings:

- Formal properties: how many and what kind of symbols can appear on the RHS?
- Linguistic: are there grammatical constructions that can be captured by one type of grammar, but not the other?

- Computational: what kind of abstract machine can correctly process the two types of grammar?

Note: answers should explain the relevance to NLP, as these formalisms have other applications in general Computer Science. As in previous years, some students gave excellent answers but others often showed imperfect understanding, and/or consisted of partly digested CS-oriented accounts. Many answers ignored the requirement to use natural language examples.

ii. This is also the kind of question that frequently comes up in exams. Some students gave excellent answers though many lost marks for one or more of the reasons discussed below, and a small number did not attempt this part of the question.

For full marks, rules should be maximally general and concise, rather than ad hoc and tailored to particular examples, should not over-generate, and answers should be **explained** as stated in the preamble. 'Over-generate' means that the grammar will accept sequences which are not grammatical sentences, such as **The boy will burgers* which might result from treating *will* as a main verb.

An example of an 'ad hoc' analysis would be to break down example (5) as *(S (S The girl likes sweet things so she will eat cake) or (NP candy))* postulating a rule *S → S or NP:* this analysis does not really make sense semantically, as "candy" is not a statement in its own right. Generally, connectives like *and, or, but* will conjoin constituents of the same syntactic category.

Many students did not attempt to explain which sentences required context-free rules, even though some of them are very similar to the examples which were used to motivate CF grammars in the subject guide.

b. This question concerned finite state machines (FSMs). This formalism should be familiar to Computing students, even if this particular application of it is not, and the problems are really quite straightforward if you work through them carefully. Most students got reasonably good marks, some excellent, though a small number did not attempt this part. Note: owing to an editing error, what was intended to be (ii) appeared with the underlined words missing. Below is how it should have appeared:

(ii) How long is the longest sentence it will accept that does not repeat any of the personal names <u>or common nouns</u>?

Due to this, an acceptable answer would have been that there is no maximum length. Credit was given for either this answer, or the intended answer which was 16. Sub-question (iv) required students to write out a formal grammar with equivalent coverage to the FSM. This was marked along similar line to question (a. ii) with credit given for more compact solutions.

c. These questions involves **regular expressions** (REs). Most students seem to be quite confident with this formalism and gave good answers, though some overlooked the requirement to list only words longer than 12 characters in (iv).

Question 2

Corpora and basic text analysis

a. This question dealt with stemming and lemmatising.

i. This was essentially a matter of reading, understanding and summarising the references provided ("bookwork"). For full marks, answers should include detailed reference to the sources they consulted (including equivalent materials beyond those listed in the assignment, if applicable). Most students did fairly well and some gave excellent answers.

ii. This question required students to discuss 10 examples that illustrated the differences between different stemmers and a lemmatiser. Credit was given for appropriately selected examples: students should keep in mind that "discuss" does not mean "list". Results may differ according to the precise release of NLTK, as a fairly recent change to the way the Porter stemmer is implemented normalises all capitalised forms to lower case. Some students complained that the results read like "gibberish", which suggested that they didn't get the point of stemming and had not studied the required readings particularly carefully. NB: answers should have been chosen to show different results where possible; while Porter and Snowball give almost identical results, there are many differences between Porter and Lancaster.

iii. This question asked student to identify any errors in the output of the applications. Given that stemming is an algorithmic process based on the form of word endings, and does not apply any grammatical or lexicographic knowledge, it is highly probable that wrong decisions will be made in some instances. If any "errors" are identified the answer must explain why they are considered to be wrong, not just assert it.

b. This question involved some rudimentary quantitative analysis of two literary texts. The techniques for solving these problems could be found in the subject guide and the NLTK book. Part (i) was an application of Conditional Frequency Distribution. The exercise was intended to show that quite simple analyses can still yield useful information, and was also meant to give students some experience in working with raw text. Generally we need to clean up and pre-process text, to make sure results are not polluted with irrelevant or uninformative material. This is why (ii) instructed you to exclude stop words and punctuation, though some students disregarded this advice; good answers would also have removed editorial matter belonging to the Gutenberg environment rather than the original text. Credit was given for any sensible answer to (iv). Most students did reasonably well and some got excellent marks. A few lost marks through failing to follow instructions to clean up the texts or explain their answers. A small number of students did not attempt this question.

Coursework commentary 2017–2018

CO3354 Introduction to natural language processing

Coursework assignment 2

Learning outcomes

This coursework assignment addressed learning outcomes 1, 3 and 4 from the subject guide:

1. utilise and explain the function of software tools such as corpus readers, stemmers, taggers and parsers
3. critically appraise existing Natural Language Processing (NLP) applications such as chatbots and translation systems
4. describe some applications of statistical techniques to natural language analysis, such as classification and probabilistic parsing.

General remarks

- You should list all references at the end of your work and they should be properly cited whenever referred to. Answers that consist largely of quoted material are unlikely to get high marks, even if properly referenced.
- You should **explain** your answers and show working (where applicable) for full marks. **Some students ignored this requirement even though it has been repeatedly stressed in examiners' commentaries. Marks were deducted accordingly.**
- You are required to submit your answers as a PDF file, with the option of including Python code in a separate Jupyter notebook. Marks may be deducted if you do not submit your work in the required format. Students are assessed both on the quality of the submitted code as a solution to the specified problems, and on how well the work is explained.

Comments on specific questions

Question 1

Classification

a. This question was adapted from an exercise in the NLTK book. Students were advised to read the Supplements to the subject guide before attempting this question, which describe some changes in the behaviour of built-in NLTK functions under NLTK 3. Students were required to follow clearly documented procedures and compare the performance of three different classifiers, and to compare the results obtained using different feature extractors.

Some students gave excellent answers, most got respectable marks though a small number did not attempt the question. Some students lost marks through avoidable errors including:

- Testing on the training set: results obtained will not be useful.
- Not understanding what is meant by a "feature extractor": this is a process which aims to identify the most relevant characteristics of a dataset in support of machine learning. Some possibilities include:

eliminating stop words, changing the size of the featureset. Credit was given for any sensible, well-motivated technique whether or not significant results were achieved. NB: students should not have expected to get the same results as their peers, since the procedure includes a random shuffle of the data.

- Giving a general description of the techniques rather than addressing specific problems.
- Not including their main results in the body of their answers in the PDF file, as instructed.

b. For this question students were required build a classifier which distinguishes fiction from non-fiction in the Brown corpus, to compare the performance of different classifiers, and to test one of the classifiers on a non-Brown text.

   Some students gave thorough, well-motivated answers though quite a few lost marks by:

   - failing to adequately explain their answers;
   - giving incomplete accounts of their results;
   - writing obscure or unnecessarily complicated code;
   - not following the instructions (e.g. listing 20 most informative features and discussing why they are informative).

Question 2

Information Extraction

This question was to do with "chunking", a process which partially parses linguistic data in order to identify items which may be relevant for information extraction. The task included evaluating a basic grammar from the NLTK book and attempting to improve its coverage by modifying the grammar rules or adding new ones. The assignment included some suggestions for extending the grammar.

Some students showed good understanding of the problem and submitted thoughtful, well presented answers with good results. Others did not attempt this question, or lost marks for various reasons:

- Lack of explanation/discussion;
- Proposed grammar rules were lacking in linguistic motivation or "hacky";
- Results not adequately reported;
- No code submitted, or evidence for claimed results;
- Some grammars massively over-generated; while precision/recall on the corpus texts were good, they would also give spurious analyses for garbage data.

Question 3

Evaluating Chatbots

This question involved evaluating the performance of software 'bots' attempting to hold a 'natural' conversation. To focus the problem more narrowly, the data consisted of responses to a series of predefined questions, rather than a free-flowing dialogue. This was sufficient to throw up a number of areas where human-level facility and fluency in conversation is hard to replicate. Many of the problems went beyond grammatical knowledge and involved "real-world" knowledge, reasoning, or maintaining a "discourse model" to keep track of references back to things that had been mentioned in a previous sentence. Most systems seemed able to cope with purely linguistic/grammatical problems, but they struggled to various extents with the more

AI-type issues. It appeared that some of the systems had been "primed" with topical knowledge, as for example many of them were able to respond in a meaningful way to the question *What do you think of Trump*?

Answers varied considerably in quality: some were thoughtful, well-expressed and motivated, showing good understanding of linguistic aspects and/or evidence of independent reading, while many were rather too short, lacking in focus, making general observations rather than focusing on concrete examples, or showing little specialist knowledge beyond that of a layperson or New Scientist reader.

Credit was given for: showing relevant technical knowledge, quality of argumentation, focused discussion of relevant examples, and clarity of expression. Please note, when referring to examples, it is more helpful to quote the actual sentences rather than simply identify them by number.