COMP 3225

Natural Language Processing

Introduction to Coursework

Stuart E. Middleton

sem03@soton.ac.uk

University of Southampton

Overview

- Assignment - Task
- Marking Scheme
- Assignment - Submission and Marking
- FAQ
- Supporting Content

Assignment - Task

- Tasks
  - Task 1 - Regex to extract chapter headings from whole book
  - Task 2 - Regex to extract every question from chapter of book
  - Task 3 - CRF named entity recognition model to extract a set of entity types from chapter of book
  - Task 4 - Regex and CRF named entity recognition model to list all characters in chapter of book

- Walkthrough of assignment PDF

Marking Scheme

- Walkthrough of marking scheme PDF

Assignment - Submission and Marking

- Local testing of your code prior to submission
  - Setup a VM **identical** to the test harness VM which submission uses
  - Request a ECS VM
    - https://sotonproduction.service-now.com/serviceportal?id=sc_cat_item&sys_id=f7d5bcbedb4ddb006f3df57eaf961924
    - Linux - RedHat Enterprise 8
    - Access from University Network only
    - Duration until end of academic year
  - Setup your VM

```
sudo yum -y install gcc gcc-c++ python39-devel
sudo yum install python39 python39-pip
sudo python3.9 -m pip install --upgrade pip
sudo python3.9 -m pip install -U nltk
sudo python3.9 -m pip install numpy
sudo python3.9 -m pip install scipy
sudo python3.9 -m pip install sklearn
sudo python3.9 -m pip install sklearn_crfsuite
sudo python3.9 -m pip install pandas
python3.9
         import nltk
         nltk.download('all')
         quit()
```

Assignment - Submission and Marking

- Local testing of your code prior to submission
  - Make a directory for testing

```
mkdir /var/lib/comp3225
```

  - Upload the assignment resources using SCP

```
cd /var/lib/comp3225

ls
>> eval_chapter.txt
>> gold_characters.txt
>> gold_ne.json
>> gold_questions.txt
>> gold_toc.json
>> nlp_submission.py
>> ontonotes_parsed.zip

unzip ontonotes_parsed.zip

ls ontonotes_parsed.json
>> ontonotes_parsed.json
```

Assignment - Submission and Marking

- Local testing of your code prior to submission
  - Replace `task<N>_submission.py` with your own code to test

```
python3.9 task1_submission.py ontonotes_parsed.json david-copperfield-book.txt
david-copperfield-chapter.txt

cat toc.json
>> {
>>   "1": "I AM BORN",
>>   "2": "I OBSERVE",
>>   "3": "I HAVE A CHANGE"
>> }

python3.9 task2_submission.py ontonotes_parsed.json david-copperfield-book.txt
david-copperfield-chapter.txt

cat questions.txt
>> And another shilling or so in biscuits, and another in fruit, eh?
>> Perhaps you’d like to spend a couple of shillings or so, in a bottle of
currant wine by and by, up in the bedroom?
>> Traddles?
>> Has that fellow’--to the man with the wooden leg--‘been here again?
```

Assignment - Submission and Marking

- Local testing of your code prior to submission
  - Replace `task<N>_submission.py` with your own code to test

```
python3.9 task3_submission.py ontonotes_parsed.json david-copperfield-book.txt
david-copperfield-chapter.txt

cat ne.json
>> {
>>   "CARDINAL": [
>>    "two",
>>    "three",
>>    "one"
>>  ],
>>  "ORDINAL": [
>>    "first"
>>  ],
v  "DATE": [
>>    "saturday"
>>  ],
>>  "NORP": [
>>    "indians"
>>  ]
>> }
```

Assignment - Submission and Marking

- Local testing of your code prior to submission
  - Replace `task<N>_submission.py` with your own code to test

```
python3.9 task4_submission.py ontonotes_parsed.json david-copperfield-book.txt
david-copperfield-chapter.txt

cat characters.txt
>> creakle
>> mr. creakle
>> mrs. creakle
>> miss creakle
```

Assignment - Submission and Marking

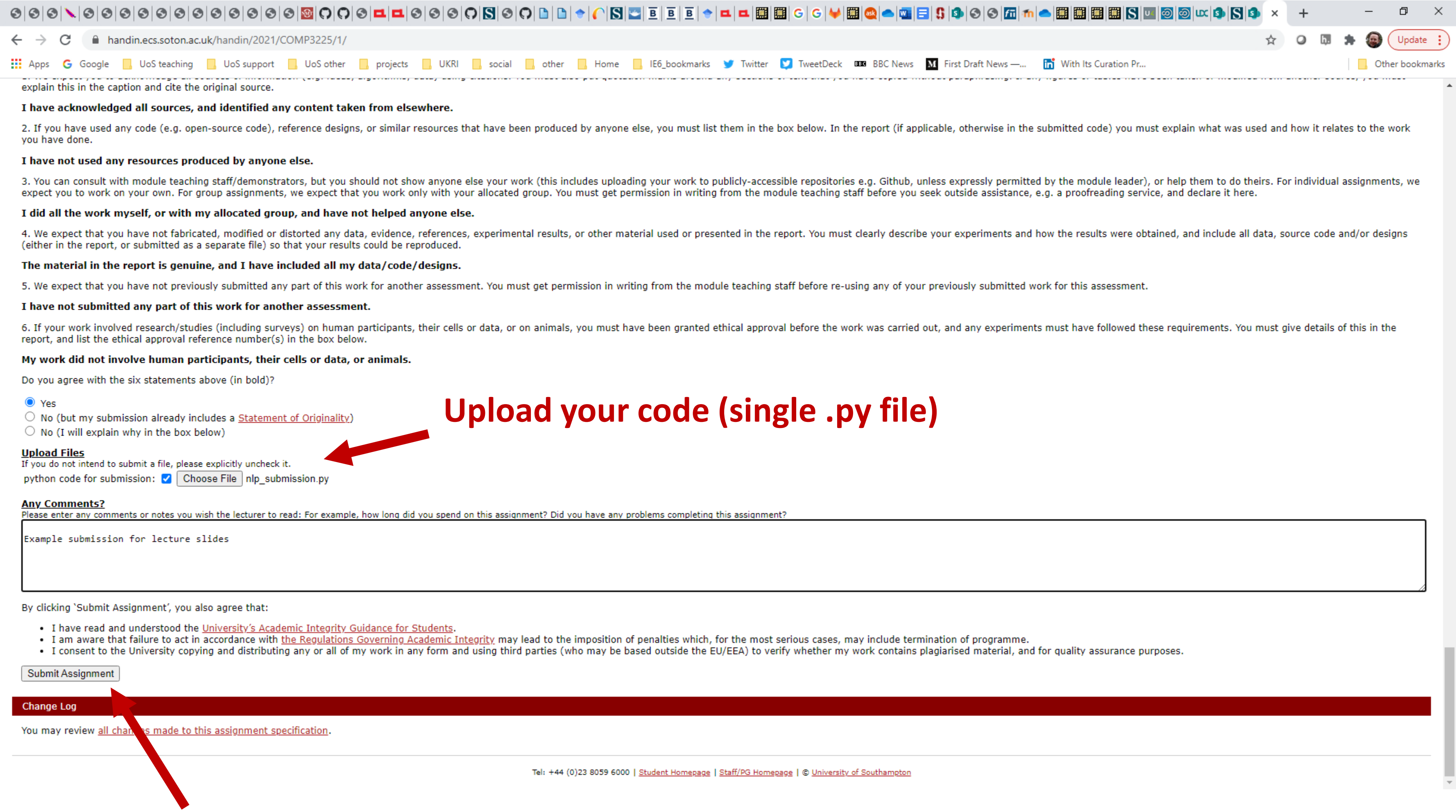
- Download 10+ books from Project Gutenberg (plain text UTF-8)
  - Web https://www.gutenberg.org/
  - Author https://www.gutenberg.org/ebooks/author/37
  - Book https://www.gutenberg.org/ebooks/766
- Create some chapter extracts for your own testing
- Make your own notes on the ground truth
  - See example ground truth files
  - gold_characters.txt
  - gold_ne.json
  - gold_questions.txt
  - gold_toc.json
- Check results on these books and chapters
  - Compute your own F1 scores for testing by comparing to ground truth files

```
python3.9 task<N>_submission.py ontonotes_parsed.json <book>.txt <chapter>.txt
```

Assignment - Submission and Marking

- Submitting your code for testing
  - You are allowed 20 submissions per task
  - Failed submission runs count against this total >> test on your own VM
- Go to the handin page for the assignment

Assignment - Submission and Marking



explain this in the caption and cite the original source.

I have acknowledged all sources, and identified any content taken from elsewhere.

2. If you have used any code (e.g. open-source code), reference designs, or similar resources that have been produced by anyone else, you must list them in the box below. In the report (if applicable, otherwise in the submitted code) you must explain what was used and how it relates to the work you have done.

I have not used any resources produced by anyone else.

3. You can consult with module teaching staff/demonstrators, but you should not show anyone else your work (this includes uploading your work to publicly-accessible repositories e.g. Github, unless expressly permitted by the module leader), or help them to do theirs. For individual assignments, we expect you to work on your own. For group assignments, we expect that you work only with your allocated group. You must get permission in writing from the module teaching staff before you seek outside assistance, e.g. a proofreading service, and declare it here.

I did all the work myself, or with my allocated group, and have not helped anyone else.

4. We expect that you have not fabricated, modified or distorted any data, evidence, references, experimental results, or other material used or presented in the report. You must clearly describe your experiments and how the results were obtained, and include all data, source code and/or designs (either in the report, or submitted as a separate file) so that your results could be reproduced.

The material in the report is genuine, and I have included all my data/code/designs.

5. We expect that you have not previously submitted any part of this work for another assessment. You must get permission in writing from the module teaching staff before re-using any of your previously submitted work for this assessment.

I have not submitted any part of this work for another assessment.

6. If your work involved research/studies (including surveys) on human participants, their cells or data, or on animals, you must have been granted ethical approval before the work was carried out, and any experiments must have followed these requirements. You must give details of this in the report, and list the ethical approval reference number(s) in the box below.

My work did not involve human participants, their cells or data, or animals.

Do you agree with the six statements above (in bold)?

● Yes
○ No (but my submission already includes a Statement of Originality)
○ No (I will explain why in the box below)

Upload Files
If you do not intend to submit a file, please explicitly uncheck it.
python code for submission: ☑ Choose File nlp_submission.py

Upload your code (single .py file)

Any Comments?
Please enter any comments or notes you wish the lecturer to read: For example, how long did you spend on this assignment? Did you have any problems completing this assignment?

Example submission for lecture slides

By clicking `Submit Assignment', you also agree that:

- I have read and understood the University's Academic Integrity Guidance for Students.
- I am aware that failure to act in accordance with the Regulations Governing Academic Integrity may lead to the imposition of penalties which, for the most serious cases, may include termination of programme.
- I consent to the University copying and distributing any or all of my work in any form and using third parties (who may be based outside the EU/EEA) to verify whether my work contains plagiarised material, and for quality assurance purposes.

Submit Assignment

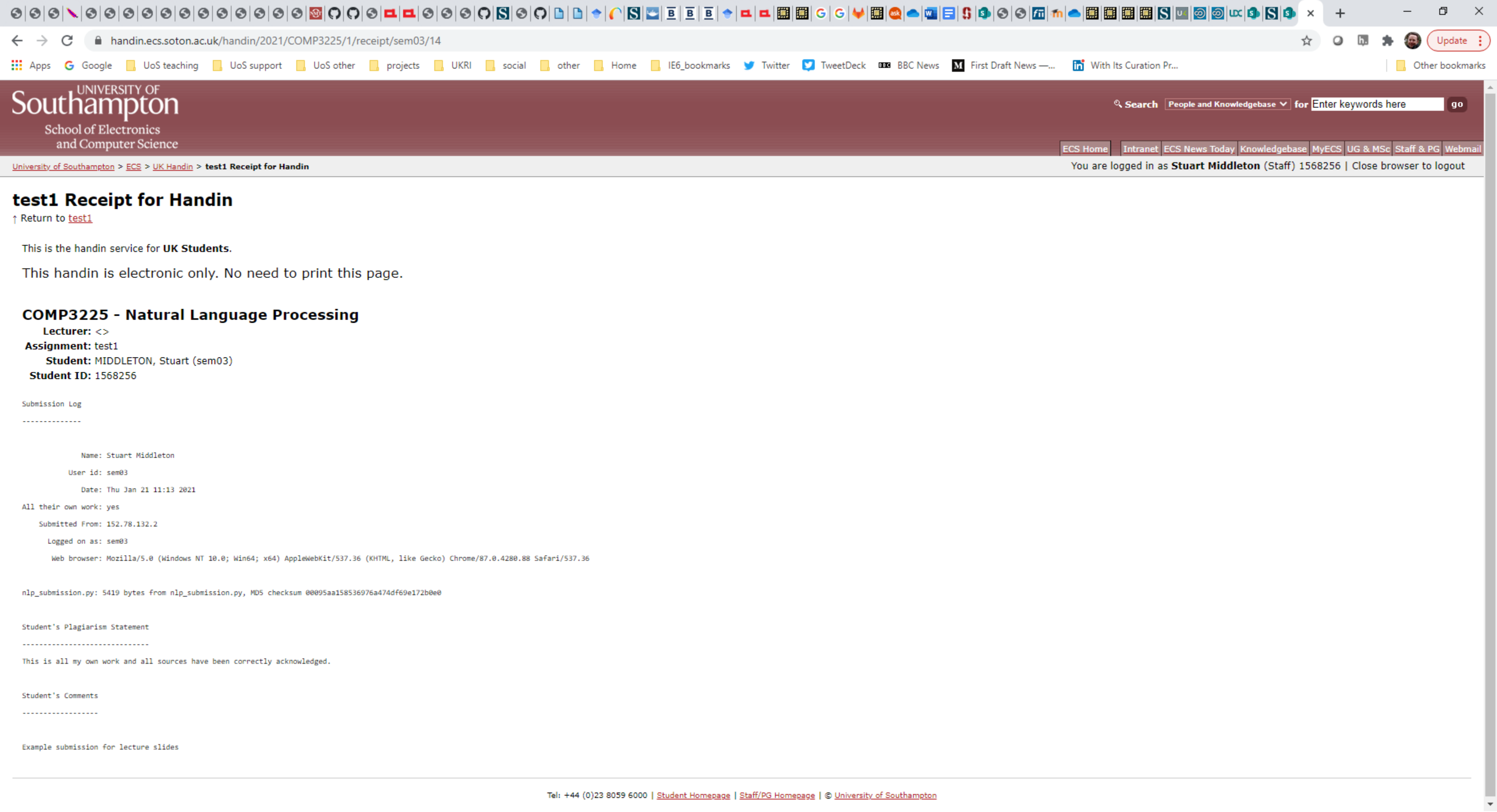
Change Log

You may review all changes made to this assignment specification.

Tel: +44 (0)23 8059 6000 | Student Homepage | Staff/PG Homepage | © University of Southampton

Submit button

Assignment - Submission and Marking



University of Southampton > ECS > UK Handin > **test1 Receipt for Handin**

You are logged in as **Stuart Middleton** (Staff) 1568256 | Close browser to logout

test1 Receipt for Handin

↑ Return to test1

This is the handin service for **UK Students**.

This handin is electronic only. No need to print this page.

COMP3225 - Natural Language Processing

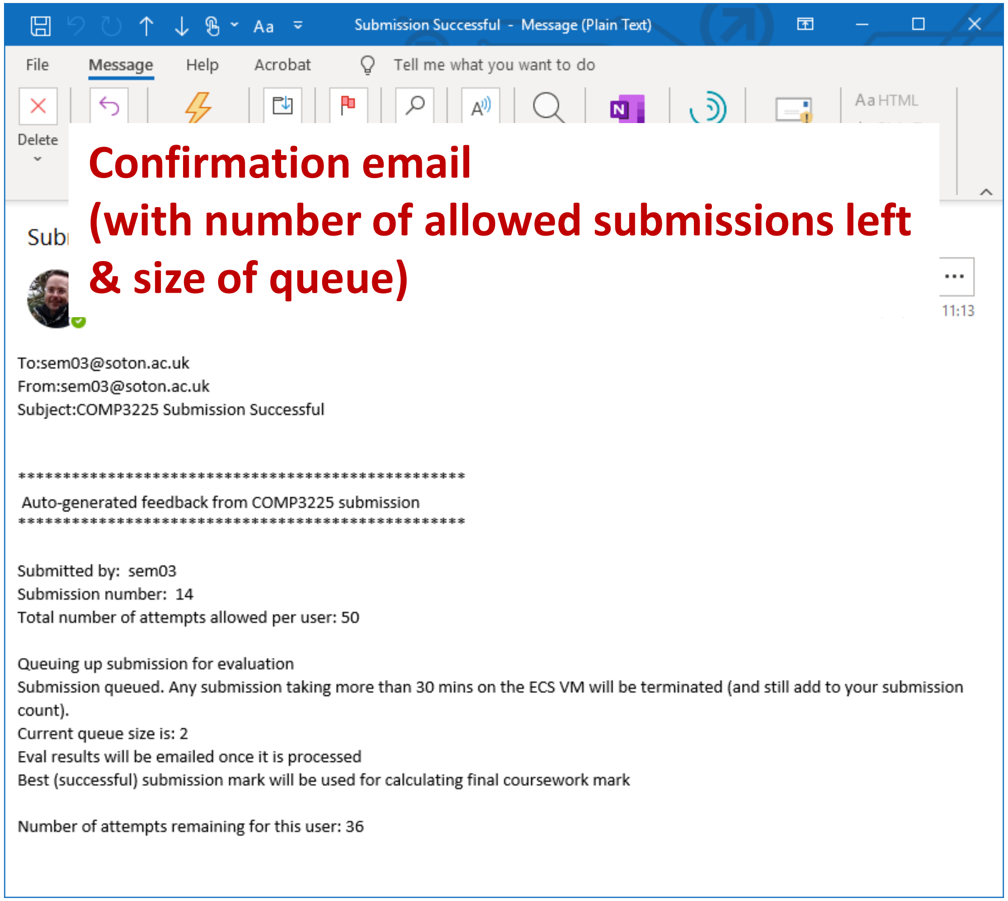
Lecturer: <>
Assignment: test1
Student: MIDDLETON, Stuart (sem03)
Student ID: 1568256

```
Submission Log

--------------


             Name: Stuart Middleton

          User id: sem03

             Date: Thu Jan 21 11:13 2021

All their own work: yes

    Submitted From: 152.78.132.2

      Logged on as: sem03

       Web browser: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/87.0.4280.88 Safari/537.36


nlp_submission.py: 5419 bytes from nlp_submission.py, MD5 checksum 00095aa158536976a474df69e172b0e0


Student's Plagiarism Statement

------------------------------

This is all my own work and all sources have been correctly acknowledged.


Student's Comments

------------------


Example submission for lecture slides
```

Tel: +44 (0)23 8059 6000 | Student Homepage | Staff/PG Homepage | © University of Southampton

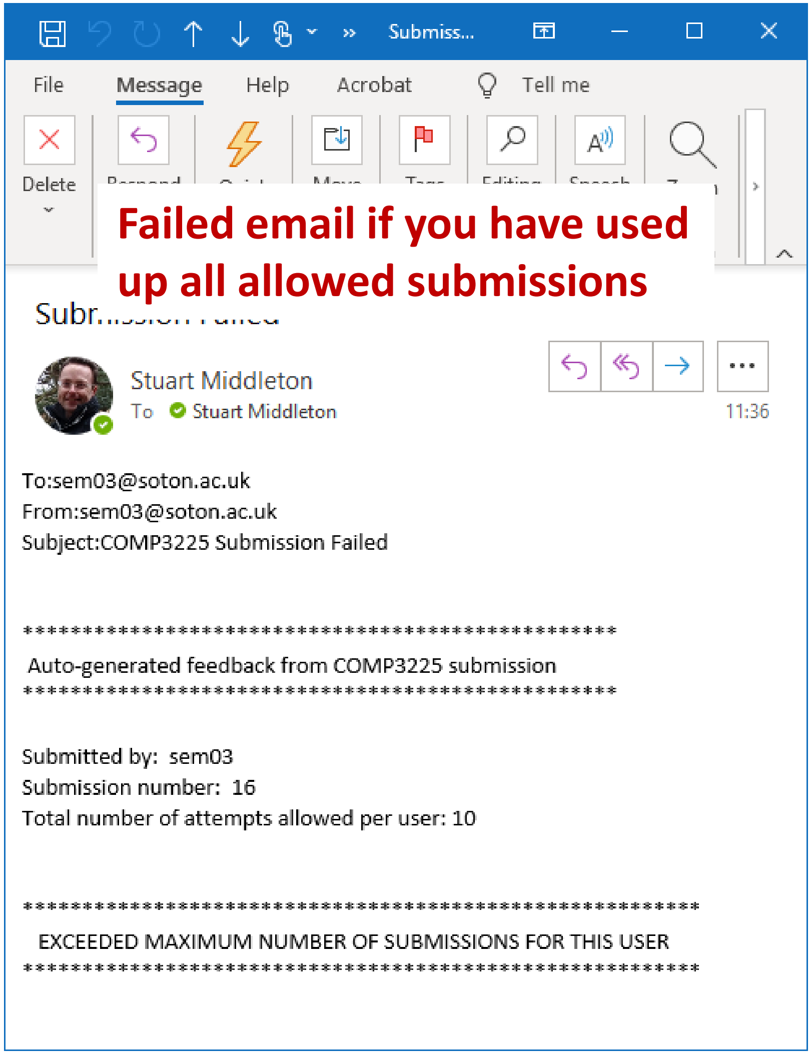
Confirmation of submission (handin) >> This is the date of submission for late penalties

Assignment - Submission and Marking


Submission Successful - Message (Plain Text)
File
Message
Help
Acrobat
Tell me what you want to do
Delete
Aa HTML
Sub
11:13

Confirmation email (with number of allowed submissions left & size of queue)

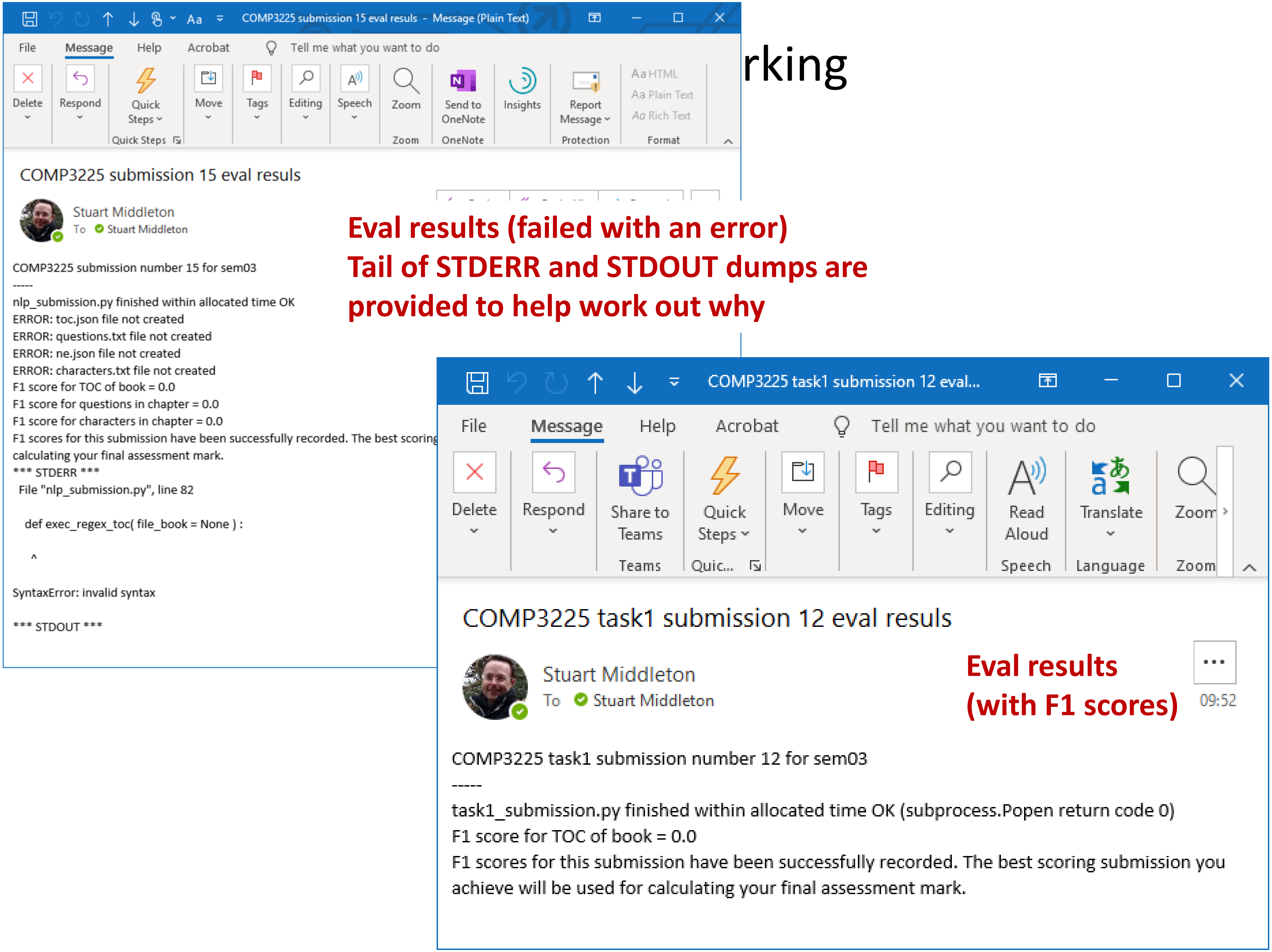
```
To:sem03@soton.ac.uk
From:sem03@soton.ac.uk
Subject:COMP3225 Submission Successful


****************************************************
 Auto-generated feedback from COMP3225 submission
****************************************************

Submitted by:  sem03
Submission number:  14
Total number of attempts allowed per user: 50

Queuing up submission for evaluation
Submission queued. Any submission taking more than 30 mins on the ECS VM will be terminated (and still add to your submission count).
Current queue size is: 2
Eval results will be emailed once it is processed
Best (successful) submission mark will be used for calculating final coursework mark

Number of attempts remaining for this user: 36
```



Failed email if you have used up all allowed submissions

```
To:sem03@soton.ac.uk
From:sem03@soton.ac.uk
Subject:COMP3225 Submission Failed


****************************************************
 Auto-generated feedback from COMP3225 submission
****************************************************

Submitted by:  sem03
Submission number:  16
Total number of attempts allowed per user: 10


***********************************************************
  EXCEEDED MAXIMUM NUMBER OF SUBMISSIONS FOR THIS USER
***********************************************************
```

rking



COMP3225 submission 15 eval resuls

Stuart Middleton
To Stuart Middleton

```
COMP3225 submission number 15 for sem03
-----
nlp_submission.py finished within allocated time OK
ERROR: toc.json file not created
ERROR: questions.txt file not created
ERROR: ne.json file not created
ERROR: characters.txt file not created
F1 score for TOC of book = 0.0
F1 score for questions in chapter = 0.0
F1 score for characters in chapter = 0.0
F1 scores for this submission have been successfully recorded. The best scoring
calculating your final assessment mark.
*** STDERR ***
 File "nlp_submission.py", line 82

  def exec_regex_toc( file_book = None ) :

    ^

SyntaxError: invalid syntax

*** STDOUT ***
```

Eval results (failed with an error)
Tail of STDERR and STDOUT dumps are provided to help work out why

COMP3225 task1 submission 12 eval resuls

Stuart Middleton
To Stuart Middleton

Eval results (with F1 scores)

```
COMP3225 task1 submission number 12 for sem03
-----
task1_submission.py finished within allocated time OK (subprocess.Popen return code 0)
F1 score for TOC of book = 0.0
F1 scores for this submission have been successfully recorded. The best scoring submission you
achieve will be used for calculating your final assessment mark.
```

FAQ

- Do submissions that fail to evaluate count to my 20 allowed attempts?
  - Yes - test your code locally to avoid wasting attempts on python errors
  - Submitting code that fails is the students fault. Lecturers will not investigate why it failed.
  - Every student has the same number of allowed submissions to be fair
  - The ONLY reason for increasing student submission limit is if the evaluation test harness code fails and submissions are lost (i.e. its not the students fault). Submissions are logged so test harness failures can be fixed.
- Python 3.9 and Red Hat are old. Can I use Python 3.10+?
  - No - you must use the exact setup of the evaluation VM
  - In future years whenever iSolutions upgrades its VM setup then the coursework requirements will be updated also. Not for this year.

FAQ

- What if I submit to handin before the coursework deadline, but the evaluation happens after the deadline?
  - Coursework **submission dates are taken from handin ONLY**. Late penalties are applied only if work is submitted to handin after the deadline.
  - Normal special consideration and extension request rules apply
  - If 100 students submit just before the deadline, and they all take the maximum 30 minutes to run, then it will take 100 x 30 minutes = 3,000 minutes = 2 days before they are all evaluated
- I have not received an evaluation email - shall I email the lecturers ASAP to ask why and chase it up
  - No - wait 2 days for the email to turn up. Emails might be delayed because the evaluation queue is long OR the university email server has some delays sending out emails
  - After 2 full days send a single email to lecturers (module leader) detailing the below information and he will confirm submission/failure when he has time
    - Student number; Submission number; Submission date

FAQ

- My code runs OK on my VM, but timed out after 30 minutes on the eval VM. Can I re-submit for free?
  - No - it is a failed attempt
  - Make sure your code runs on your test VM in 25 minutes, to avoid getting too close to the 30 minute timeout
  - Submit a few 'banker' code runs to make sure you have some good results safely recorded before trying to push the limits on runtime
- Can I dump the hidden book and chapter in the STDOUT
  - No - this is cheating and will trigger an academic integrity issue
  - If you somehow dump book or chapter >> **tell lecturer team immediately**
  - Advice >> **remove all print statements** from submission code
- Can my code work write data outside the current working directory?
  - No - writing files outside the current working directory is not required and might corrupt the automated test harness setup
  - If code is found to write files outside the current working directory in a malicious way or attempt to cheat it will trigger an academic integrity issue

FAQ

- How is F1 score computed?
  - Gold truth files are compared to generated files from submission
  - Task1 >> original case, strip() whitespace, exact text match
  - Task2 >> original case, strip() whitespace, exact text match
  - Task3 >> lowercase, strip() whitespace, exact text match
  - Task4 >> lowercase, strip() whitespace, exact text match
  - Symbols and spaces count must match - return EXACTLY the same text as the original book or chapter plain text UTF-8 file
- **Example**

chapter.txt

‘I am a determined character,’ said Mr. Creakle. ‘That’s what I am. I
do my duty. That’s what I do. My flesh and blood’--he looked at Mrs.
Creakle as he said this--‘when it rises against me, is not my flesh
and blood. I discard it. Has that fellow’--to the man with the wooden
leg--‘been here again?’

task 2 gold truth question.txt

Has that fellow’--to the man with the wooden leg--‘been here again?

(1) In Gutenberg books plain text UTF-8 files are serialized using a column width. You need to repair the Gutenberg text file you are given to run with so that sentences are not split by newlines

(2) Exact text extractions are expected, so we expect Has that fellow’--to the man with the wooden leg--‘been here again? NOT Has that fellow been here again? NOT Has that fellow--to the man with the wooden leg--been here again?

FAQ

- Play fair
    - All submitted code is recorded via handin, and stored to allow academic integrity checks
    - A manual check of code will be performed to scan for cheating attempts
    - Any attempt at cheating or malicious behaviour will trigger an academic integrity issue
    - Don't cheat - the academic integrity penalties far outweigh any potential rewards!

Supporting Content

- Lectures
  - Lecture 3 Regular Expressions
  - Lecture 7 Named Entity Recognition
- Labs
  - CRF Named Entity Recognition lab
  - Available on module wiki

Questions

- Panopto Quiz - 1 minute brainstorm for interactive questions

  Please write down in Panopto quiz in **1 minute** two or three questions that you would like to have answered at the next interactive session.

  Do it **right now** while its fresh.

  Take a screen shot of your questions and **bring them with you** at the interactive session so you have something to ask.