Distant Reading (VT-23)
Course coordinator Matti La Mela, matti.lamela@abm.uu.se

Lab assignment week 3

Do the following exercises in a Jupyter Notebook file (see T0), submit the file in Studium by Monday 6 February at 23:59.

Copy-paste the individual assignment "tasks" (T1, T2, T3, T4…) in Markdown cells. For programming questions, answer below in a code cell. In the code cell, use comments with hashtag where needed. Tasks that are "open questions" without coding are answered by typing your answers (you can add images).

Double-check that your python code works. Feel free to reuse as much code from the learning materials as you like, use the tutorial videos, search solutions on the internet and look for help in the reference readings. If you remain stuck or have questions, you can use our slack workspace to ask more help.

NB. Ts marked with optional are required for reaching grade VG, but are not compulsory.

Assignments:

T0) Create a new Notebook file in Jupyter Lab. Name the file yourlastname_lab3.ipynb (e.g. La_Mela_lab3.ipynb). Copy-paste the individual assignment "questions" (Ts) in Markdown cells and your answer in a Markdown / code cell below this depending on the question.

1. NER with spacy (Materials 3b, first part)

T1) Read the txt-file "Verne_Around_the_World.txt" into a string. The txt contains the book "Around the World in Eighty Days" (by Jules Verne; downloaded from gutenberg.org). (The book starts from chapter 1, and has been processed with split() and " ".join() for removing line breaks and whitespaces.)

T2) Import spacy and the English language model. Use spacy nlp() to process the text into a doc object, which you can name "text_doc". (You can only process first 50000 characters of the book to be a bit faster).

T3) Create a list "persons_list = []". Use a for-loop to iterate all the named entities (text_doc.ents), and store all the person entities (entity.label == "PERSON") to the list. Print the persons_list.

T4) Use Counter.most_common() (from collections) to count the top five person names in your list of person names.

T5) Use "displacy" in spacy to visualize all the named entities in the first 5000 characters of text_doc. Do you find any incorrect or questionable NE classifications made by spacy?

OPTIONAL T6) There is also a POS tag for proper nouns, "PROPN", that has a lot in common with named entities. Pick one of the top person names in the novel (T4), and count all the cases where spacy has identified that name as a proper noun ("PROPN"). Discuss: when might POS tags be a better choice than using NE classes? For what kind of questions does only NER work?

Tip: Iterate the text_doc in a for-loop to do the counting

2. Reading multiple files (Materials 3a)

T7) Import spacy and the English language model.

T8) Use glob to create a list with all the filenames (*.txt) in the directory "selection_DHQ". Print the number of files on your list.

T9) Write a for-loop that reads the files on your file list one by one, and does the following for every file it opens: find all PERSON entities in the file and find all GPE entities and append them to two separate lists. When you are done, print top 10 person names and top 10 geopolitical places (eg. with Counter.most_common())

OPTIONAL T10) Use glob to go through all the DHQ articles, thus, use the "DHQ_corpus_complete_2007_2020". Write a for-loop that goes through all the files: during each iteration, store all (#surnameyear) expressions from each article and save them to a list (find a good regular expression, and use re.findall() for each article, see Material_2b).

OPTIONAL T11) (Optional) Present which five works were the most cited in 2007-2020. Find out the complete reference for the most cited work (you can check the sources for "#surname" manually by opening the txt files).

3. NER analysis (Materials 3b, part two)

T12) Import spacy and the English language model.

T13) Open the file "commons_speeches_1996.txt". Split the text by "\n\n" into a list where you separate the speeches /parts of the debates in a way that Spacy can handle the elements (see Materials 3b, part two)

T14) Write a for-loop where you go through the speeches (at least 500 first speeches, it is a big file; try opening it with excel or notepad++ if you like!) and extract all organization/institution names (label_ == "ORG").

T15) Print your list and discuss: What kind of organizations or institutions you find. Could there be use for some stopwords?

T16) Think what metadata about the speeches / speakers would be useful for you, and how it could be used to further your analysis (i.e. what useful information could there be available related to these speeches)?

OPTIONAL T17) Extract place names (GPE and other locations) from the parliamentary debates in 1996 and 2010 (take 1000 speeches at least). What differences do you find in the top 10 place names for these years? The speeches from 2010 are in the file commons_speeches_2010.txt.

OPTIONAL T18) Start with the debates held in 1996: Pick one country name from your list (T17). Find all occurrences of the country in the speeches (using spacy tokens) and store all sentences (.sent.text) where the term appears to a list. Join the list into a string (" ".join()). Process the string (with sentences) with spacy and store all non-stopword nouns from these sentences. What are the most common nouns?

OPTIONAL T19) Repeat the same with the debates from 2010. Do you see any differences?

OPTIONAL T20) Discuss briefly what other analysis could be done with the sentences that spacy extracts (with this kind of material).