QuotationTool

In this notebook, you will use the *QuotationTool* to extract quotes from a list of texts. In addition to extracting the quotes, the tool also provides information about who the speakers are, the location of the quotes (and the speakers) within the text, the identified named entities, etc., which can be useful for your text analysis.

Note: This code has been adapted (with permission) from the GenderGapTracker GitHub page and modified to run on a Jupyter Notebook. The quotation tool's accuracy rate is evaluated in this article.

> **User guide to using a Jupyter Notebook**
> If you are new to Jupyter Notebook, feel free to take a quick look at this user guide for basic information on how to use a notebook.

1. Setup

Before you begin, you need to import the QuotationTool and the necessary libraries and initiate them to run in this notebook.

In [1]:
```
# import the QuotationTool
from extract_display_quotes import QuotationTool, DownloadFileLink

# initialize the QuotationTool
qt = QuotationTool()
```

```
[nltk_data] Downloading package punkt to /home/jovyan/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
Loading spaCy language model...
This may take a while...
Finished loading.
```

> **Installing Libraries**
> The requirements file **environment.yml** is included with this notebook. Take a look inside to find out what libraries you have just installed with the above command.

2. Load the data

This notebook will allow you to extract quotes directly from a text file (or a number of text files). Alternatively, you can also extract quotes from a text column inside your excel spreadsheet (see an example here).

Uploading your text files
If you have a large number of text files (more than 10MB in total), we suggest you compress (zip) them and upload the zip file instead. If you need assistance on how to compress your file, please check the user guide for more info.

Large file upload
If you have ongoing issues with the file upload, please re-launch the notebook via Binder again. If the issue persists, consider restarting your computer.

In [2]:
```
# upload the text files and/or excel spreadsheets onto the system
display(qt.upload_box)
print('Uploading large files may take a while. Please be patient.')
print('\033[1mPlease wait and do not press any buttons until the progress bar a
```

```
VBox(children=(FileUpload(value=(), accept='.txt, .xlsx, .csv, .zip', descript
ion='Upload your files (txt, csv…
Uploading large files may take a while. Please be patient.
Please wait and do not press any buttons until the progress bar appears...
```

Once your files are uploaded, you can see a preview of the text in a table format (pandas dataframe).

Tools:

- nltk: for sentence tokenization
- spaCy: for text cleaning and normalisation
- pandas: for storing and displaying in dataframe (table) format

Specify the number of rows to display
By default, you will preview the first 5 rows of the extracted quotes in a pandas dataframe (table) format. However, you can preview more or less rows by specifying the number of rows you wish to display in the variable 'n' below.

In [3]:
```
# specify the number of rows you wish to display
n=5

# display a preview of the pandas dataframe
qt.text_df.head(n)
```

Out[3]:

	text_name	text	text_id
0	5c1548a31e67d78e2771624f	Looking for a job, kid? Try a store, a bank, a...	f035f5d8e405fc4d7c36b2c7759f4730
1	5c489df91e67d78e271d66c5	After fading down the stretch last season, the...	81bebde107bc12c7cb15c69f9ae5ebaf
2	5c28972a795bd2fac69fa974	The federal Liberal party has selected the own...	666e6b9312bd832517c5b16df355460b
3	5c29beda1e67d78e27b74939	Australia is expressing concerns over the case...	4787aa6578df43210721ee6a426e1362
4	5c182ac21e67d78e277944ad	Darian Lonechild is new to advocacy work, but ...	fe61078e97e0777d48a3285d272018b0

3. Extract the quotes

Once your texts have been stored in a pandas dataframe, you can begin to extract the quotes from the texts. You can also extract named entities from your text by setting the named entities you wish to include in the below *inc_ent* variable. If you are extracting quotes from a lot of texts, be patient. As a guideline, for a corpus with a file size of 54.13 MB (~26,000 newspaper articles in plain text format), it can take ca 45 minutes to extract quotes.

Tools:

- quote_extractor: for extracting quotes and speakers
- spaCy: for extracting named entities

Specify the number of rows to display
By default, you will preview the first 5 rows of the extracted quotes in a pandas dataframe (table) format. However, you can preview more or less rows by specifying the number of rows you wish to display in the variable 'n' below.

Memory limitation in Binder
The free Binder deployment is only guaranteed a maximum of 2GB memory. Processing very large text files may cause the session (kernel) to re-start due to insufficient memory. Check the user guide for more info.

In [4]:
```
# specify the named entities you wish to include below
inc_ent = ['ORG','PERSON','GPE','NORP','FAC','LOC']

# specify the number of rows you wish to display
n=5

# extract quotes from the text and preview them in a pandas dataframe (table) f
quotes_df = qt.get_quotes(inc_ent)
```

```
# display a preview of the pandas dataframe
quotes_df.head(n)
```

```
Extracting quotes...
This may take a while...
100%|██████████| 5/5 [00:01<00:00,  4.08it/s]
```

Out[4]:

	text_id	text_name	quote_id	quote	quo
0	f035f5d8e405fc4d7c36b2c7759f4730	5c1548a31e67d78e2771624f	0	, the retail sector in Ontario employed 226,00...	(2
1	f035f5d8e405fc4d7c36b2c7759f4730	5c1548a31e67d78e2771624f	1	Those sectors are the best bets for youth look...	(3
2	f035f5d8e405fc4d7c36b2c7759f4730	5c1548a31e67d78e2771624f	2	"Some of those 'soft' skills are in short supp...	(7
3	f035f5d8e405fc4d7c36b2c7759f4730	5c1548a31e67d78e2771624f	3	they're what employers are looking for	(7
4	f035f5d8e405fc4d7c36b2c7759f4730	5c1548a31e67d78e2771624f	4	the unemployment rate of young Canadians is mo...	(9

What information is included in the above table?

In general, the quotes are extracted either based on syntactic or heuristic rules. Some quotes can be stand-alone in a sentence, or followed by another quote (floating quote) from the same speaker. Please refer to this document for further information about the quote extraction process.

text_id: the unique ID of the text.

text_name the name of the text, i.e., the name of the .txt files or the 'text_name' column in the excel spreadsheet.

quote_id/speaker_id: the unique ID of the extracted quote/speaker.

quote/speaker: the content of the extracted quote and the speaker.

verb: the verb used to determine the extracted quote.

quote_index/speaker_index/verb_index: the location of the first and the last characters of the extracted quote/speaker/verb in the text.

quote_entities/speaker_entities: the entity name and type of the entities identified in the extracted quote/speaker.

quote_token_count: the length of the extracted quote (in character).

quote_type: the type of quote based on how it is extracted.

floating_quote: whether the extracted quote is a floating quote, i.e., a follow up quote from the same speaker (The value TRUE here means that the quote is a floating quote, while FALSE means that the quote is not a floating quote).

Quotation symbols: Q (Quotation mark), S (Speaker), V (Verb), C (Content).

Named Entities: PERSON (People, including fictional), NORP (Nationalities or religious or political groups), FAC (Buildings, airports, highways, etc.), ORG (Companies, agencies, institutions, etc.), GPE (Countries, cities, states), LOC (Non-GPE locations, mountain ranges, bodies of water).

4. Display the quotes

Once you have extracted the quotes, you can see a preview of the quotes using spaCy's visualisation tool, displaCy.

Tools:

- displaCy: for displaying quotes, speakers and named entities
- ipywidgets: for interactive tool

Select the text and the entities to show
In order to preview the extracted information, select the text you wish to analyse and which entities to show. Then, you can click the *__Preview__* button to display them and the *__Save Preview__* button to save them as an html file.

In [5]:
```
# display a preview of the extracted quotes, speakers and entities within the t
qt.analyse_quotes(inc_ent)
```

Out[5]:
```
VBox(children=(HBox(children=(VBox(children=(HTML(value='<b>Select which entit
y to show:</b>', placeholder='')…
```

Select the text and the entities to show

You can also display the top named entitites identified in the quotes and/or speakers. You just need to select the text to analyse (option to analyse 'all texts' is also available), whether to display the identified entities in the speakers and/or quotes, whether to display the entity names and/or types, the number of top entities to display and finally, click the ***Show Top Entities*** and ***Save Top Entities*** buttons to display and save them, respectively.

In [6]:
```
# check the top named entities identified in the quotes and/or speakers
qt.analyse_entities(inc_ent)
```

Out[6]:
```
VBox(children=(HBox(children=(VBox(children=(HTML(value='<b>Select which entit
y to show:</b>', placeholder='')…
```

Capitalized words

Please note that lowercase or UPPERCASE words such as quote, QUOTE, Quote, etc. are recognised as different words by the tool, so you may see that they are counted differently in the above analysis.

5. Save the quotes

Finally, you can run the below code to save the quotes pandas dataframe into an Excel spreadsheet and download them to your local computer.

In [7]:
```
# specify output directory and file name
output_dir = './output/'
file_name = 'quotes.xlsx'

# save quotes_df into an Excel spreadsheet
from pyexcelerate import Workbook
values = [quotes_df.columns] + list(quotes_df.values)
wb = Workbook()
wb.new_sheet('Sheet1', data=values)
wb.save(output_dir + file_name)

# download quotes_df to your computer
print('Click below to download:')
display(DownloadFileLink(output_dir + file_name, 'quotes.xlsx'))
```

```
Click below to download:
```
quotes.xlsx

In [ ]:
```
```