README

Author: Muzammil Ahmed

1. Setup the environment

```
$ python -m pip install virtualenv
$ virtualenv venv
$ . venv/Scripts/activate  # you should now see a (venv) in your cli
```

2. Install the package

Install the package using:

```
$ python -m pip install -r requirements.txt
```

3. Usage

To generate the output.json, run the following from root:

```
$ python code/gen-unified-corpora.py
```

To view the statistical metrics, start the jupyter lab as follows:

```
$ jupyter lab
```

And then navigate to `code/statistical-analysis.ipynb`

4. Explanation of Specific Words Determination

First of all a couple of points on what was considered a "word":

1. One token (i.e. from spaCy's tokenizer) was treated as one word
2. Keep count of words only that are NOT stopwords (i.e. spaCy stopwords)

Then most specific words were defined as follows:

A word in an argument (major claim, claims or premises) may be called most-specific to that argument if it has the highest frequency in that argument AND may not appear in most-specific word list of other arguments.

For example the word "people" appears in all three lists as the word with highest frequency in their respective arguments but is not a specific word since it appears in more than one top-10 lists.

The technical implementation was achieved using three major data structures:

1. Dictionary to keep the words and their frequency count
2. Sets to check intersection among lists
3. Heap trees to fetch the next highest frequency element (within an arugment) efficiently

Also, it was important to implement the algorithm in such a way that no matter which of the three lists of most-specific words was determined first, it should have no impact on other arguments' most-specific words lists.