Name	Student ID	Contributions
Monisha Mohandas	301384925	Coding for A and B, Annotation For A
Olivia Calton	301291735	Annotations for A and Comments for A+
Mithula Barua	301217522	Writeup for B and Questions for B+

Are all members of a cluster included?
Based on our five selected files, we believe that all members of our cluster were not included. In part four we have found that the cluster was not able to identify all "PERSON". For example in the sentence found in text 1 file (5c1548a31e67d78e2771624f.txt) "A union representative has said the derailed train was shorter than the 135 cars CP has run in recent years but a veteran Boston-based engineer said 112 cars is large for a train of full grain hoppers". In this sentence "a union representative" was not identified as a person and our "veteran Boston-based engineer" was not identified as a person. This suggests that the cluster chain did not pick up these entities correctly as persons. This means that the cluster couldn't identify that person can also be described through their role or job position. Even though "a union representative" might not specifically name a person, it refers to a position of an organization. Here, "person" does not necessarily relate to a particular individual but it can also refer to any number of people who take on the job position. Similarly, "Our veteran Boston-based engineer" does not give a name entity but it is considered a person because it refers to the person's position as an engineer. The cluster chain failed to understand the context of the whole sentence and was not able to identify the entity name for it. In some examples, Persons of the same individuals were not consistent. For example, in text file five (5c1452701e67d78e276ee126) Dough McCallum was consistent in recognizing it was a name but throughout the text it missed some of it as McCallum to be identified as Person.

Are there clusters with the wrong members?
Our cluster chain for the cluster code did what it is supposed to do. From the selected five files we do not notice any wrong members. The cluster chain was able to pick on Persons very well for our five text files. However sometimes it would miss the name entity since the cluster should identified as the name entity of a person. For example in text 1 the sentence "Will Young a locomotive mechanic", this sentence for Will Young was not identified as "Person". This was strange because Will Young should have been identified as a name entity for persons. This suggests that the coherence chain lacked the ability to recognize and understand all forms of person names and couldn't classify Will Young as a person entity. We believe that this could occur because our chain was unable to identify which word should be grouped together with

their proper entities. Also, in text five for the sentence “ TransLink CEO Kevin Desmond PERSON said Thursday one of McCallum's suggestions.” In this sentence the cluster chain failed to recognize that McCallum is the name entity of a person. This suggests that it cannot recognize the name entity because of how the sentence is structured.

Are there clusters that ought to be merged?
Yes, the clusters can be merged if the same entity is mentioned in different paragraphs of a text, but ends up in different clusters (chains). We can merge similar entities together based on the function. For example, if two clusters reflect the same entity that is cited in various portions of the text, we can combine it. When a single entity is mentioned in several sentences using various noun phrases or pronouns. In such cases it will be ideal to combine and merge them to be consistent with our entity coherence. This could help with accuracy and consistency with pronouns and name entities. For example, from our examples, if we take “software engineer” we can cluster for it to filter software engineer and software programmer. We can merge these two to maintain coherence.

How could you use the coreference chains to find the speaker of the quote?
Based on our class exercise on SpaCy and coherence lecture, if we wanted to use the coreference chains to find the speech of the quotes we can do this by creating a code with a function that looks for the speaker of quotes. We would keep the same code from our A2_starter but make a function code for the speaker of the quote. We have provided an example of how this would look like in our Github repository for the extraction of the speaker of the quote. We added a code such as quote text = token.text this string will analyze the text and tokenize the sentence and ask to print the quote of text with the speaker using the quote print(f”’Quote: {quote text} - Speaker: {ent.text}”). This will give us the speaker of the code. Based on this extracted code we use a short sentence from our text file to see if it was able to get the quote and person together in a sentence. We executed the code and it gave us the speaker of the quote for the sentence.

Coherence Chains to Find Code Extraction For Quotes

1. We need to import all the coherence with
coreferee, spacy
nlp = spacy.load('en_core_web_trf')
nlp.add_pipe('coreferee'). We use the coherence chain

2. We needed to import Spacy to understand the coherence chain.
import spacy
from spacy import displacy

3. We use Text file 1 sentence to where there is a speaker with the quote. We took the text and had the sentence in text in it. We used the text to doc = nlp(text) which will use the NLP for it to process the text

4. We use the side visualization from part 4 to find the person using

```
 options = {"ents": ["PERSON"],
          "colors": {"PERSON": "lightsteelblue"}}
```

displacy.render(doc, style="ent", options=options, jupyter=True). This code will help identify the person

5. In this code we ask for the code to find the person of the code. This will get the person who says the dialogues in quotes. We took the code to make sure it will bring the speaker name when it says the quote. We use the code quote_text = token.text, this will tokenize the code of the chain and this code “print(f"Quote: {quote_text} - Speaker: {ent.text}")” will give the output for the speaker in which the quotes were been said.