Natural Language Processing ECS763P
Course Work 3: CoReference Resolution

Name : Tanmaiyii Rao
Student ID - 140361229

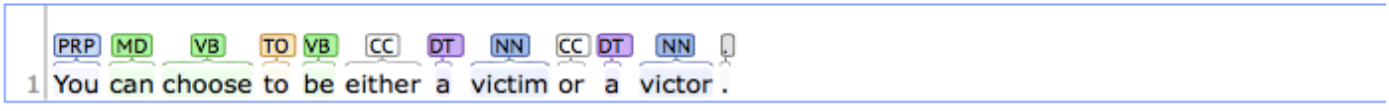
1. Coreference with the Stanford CORE NLP System

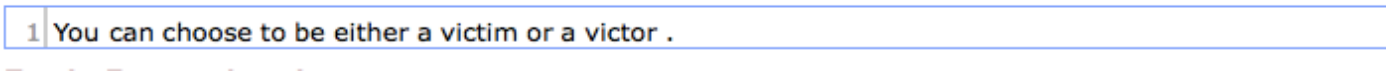
Task 1.

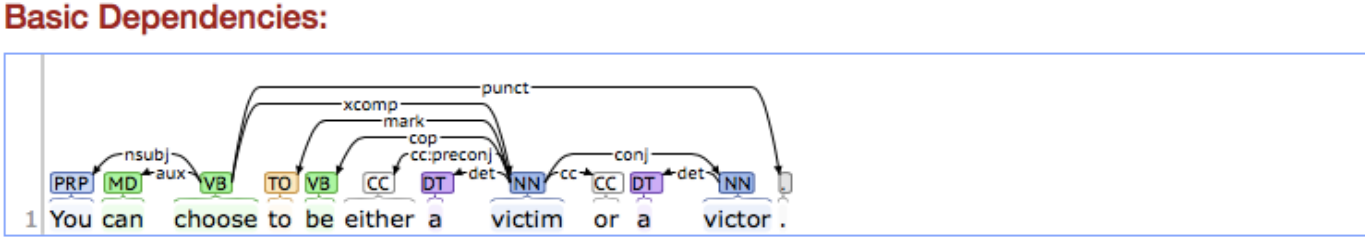
a) The types of annotators the system can carry out are : tokenize, cleanxml, ssplit, pos, lemma, ner, regexner, sentiment, truecase, parse, depparse, dcoref, relation, natlog, quote.
b) The languages it can handle are : English, Arabic, German, Chinese, French and Spanish.
c) Stanford CoreNLP provides a set of human language technology tools. It can give the base forms of words, their parts of speech, whether they are names of companies, people, etc., normalize dates, times, and numeric quantities, mark up the structure of sentences in terms of phrases and syntactic dependencies, indicate which noun phrases refer to the same entities, indicate sentiment, extract particular or open-class relations between entity mentions, get the quotes people said, etc.

Task 2.

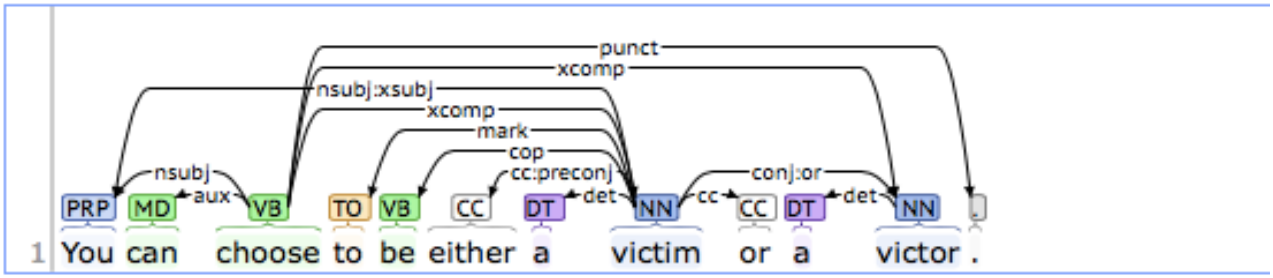
a) As can be seen, there are 4 types of recognition : Part-of-Speech, Named Entity Recognition , Ba
Dependencies and Enhanced++ Dependencies and Open Information Extraction (Open IE)



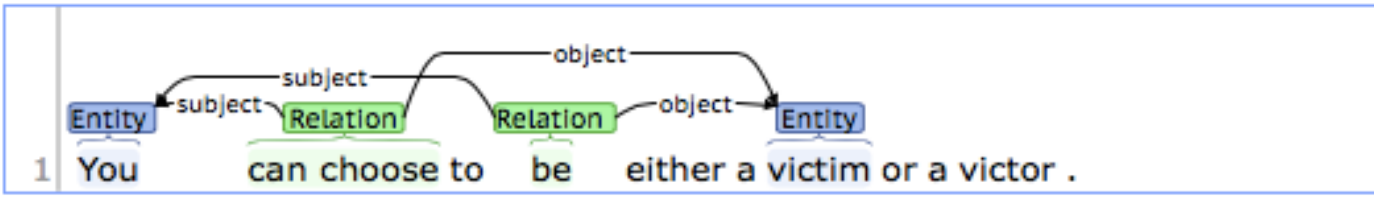




Enhanced++ Dependencies:



Open IE:



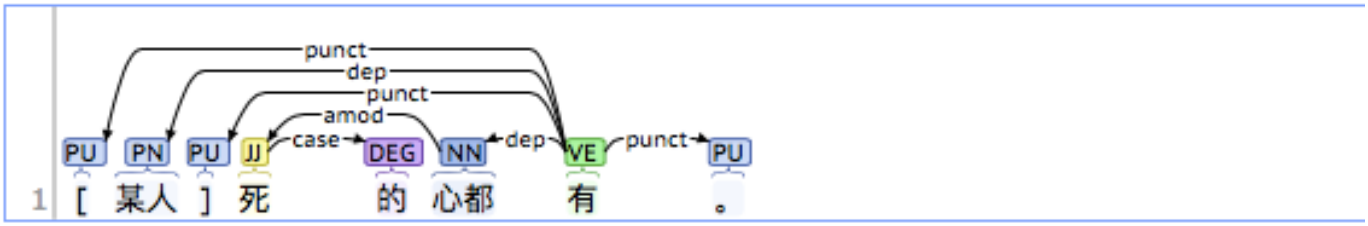
Task 3

Sentence in Chinese (Mandarin)

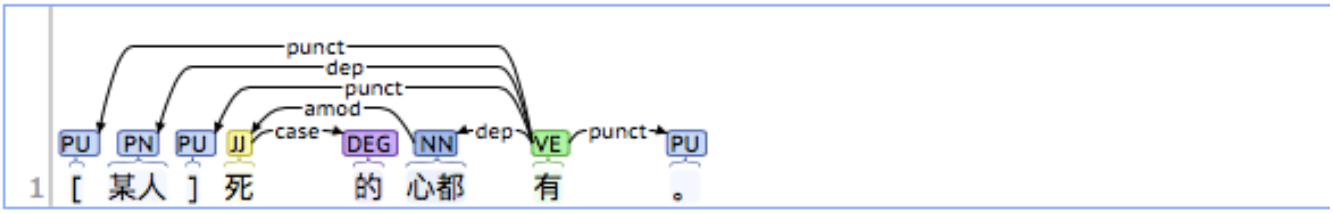
Part-of-Speech:

Named Entity Recognition:

Basic Dependencies:



Enhanced++ Dependencies:



Open IE:

Task 4: The Stanford Core NLP Package and Java version 8 have been installed.

Task 5:
Sentence #1 (6 tokens):
list me the seats to Denver

[Text=tokenize CharacterOffsetBegin=0 CharacterOffsetEnd=8 PartOfSpeech=NN Lemma=tokenize NamedEntityTag=O]
[Text=, CharacterOffsetBegin=8 CharacterOffsetEnd=9 PartOfSpeech=, Lemma=, NamedEntityTag=O]
[Text=ssplit CharacterOffsetBegin=9 CharacterOffsetEnd=15 PartOfSpeech=NN Lemma=ssplit NamedEntityTag=O]
[Text=, CharacterOffsetBegin=15 CharacterOffsetEnd=16 PartOfSpeech=, Lemma=, NamedEntityTag=O]
[Text=pos CharacterOffsetBegin=16 CharacterOffsetEnd=19 PartOfSpeech=NNS Lemma=po NamedEntityTag=O]
[Text=, CharacterOffsetBegin=19 CharacterOffsetEnd=20 PartOfSpeech=, Lemma=, NamedEntityTag=O]
[Text=lemma CharacterOffsetBegin=20 CharacterOffsetEnd=25 PartOfSpeech=NN Lemma=lemma NamedEntityTag=O]
[Text=, CharacterOffsetBegin=25 CharacterOffsetEnd=26 PartOfSpeech=, Lemma=, NamedEntityTag=O]
[Text=ner CharacterOffsetBegin=26 CharacterOffsetEnd=29 PartOfSpeech=NN Lemma=ner NamedEntityTag=O]
[Text=, CharacterOffsetBegin=29 CharacterOffsetEnd=30 PartOfSpeech=, Lemma=, NamedEntityTag=O]
[Text=parse CharacterOffsetBegin=30 CharacterOffsetEnd=35 PartOfSpeech=VB Lemma=parse NamedEntityTag=O]
[Text=, CharacterOffsetBegin=35 CharacterOffsetEnd=36 PartOfSpeech=, Lemma=, NamedEntityTag=O]
[Text=dcoref CharacterOffsetBegin=36 CharacterOffsetEnd=42 PartOfSpeech=NN Lemma=dcoref NamedEntityTag=O]
[Text=- CharacterOffsetBegin=43 CharacterOffsetEnd=44 PartOfSpeech=: Lemma=- NamedEntityTag=O]
[Text=file CharacterOffsetBegin=44 CharacterOffsetEnd=48 PartOfSpeech=NN Lemma=file NamedEntityTag=O]
[Text=input.txtList CharacterOffsetBegin=49 CharacterOffsetEnd=62 PartOfSpeech=NN Lemma=input.txtlist NamedEntityTag=O]
[Text=me CharacterOffsetBegin=63 CharacterOffsetEnd=65 PartOfSpeech=PRP Lemma=I NamedEntityTag=O]
[Text=the CharacterOffsetBegin=66 CharacterOffsetEnd=69 PartOfSpeech=DT Lemma=the NamedEntityTag=O]

[Text=seats CharacterOffsetBegin=70 CharacterOffsetEnd=75 PartOfSpeech=NNS Lemma=seat NamedEntityTag=O]
[Text=on CharacterOffsetBegin=76 CharacterOffsetEnd=78 PartOfSpeech=IN Lemma=on NamedEntityTag=O]
[Text=the CharacterOffsetBegin=79 CharacterOffsetEnd=82 PartOfSpeech=DT Lemma=the NamedEntityTag=O]
[Text=flight CharacterOffsetBegin=83 CharacterOffsetEnd=89 PartOfSpeech=NN Lemma=flight NamedEntityTag=O]
[Text=to CharacterOffsetBegin=90 CharacterOffsetEnd=92 PartOfSpeech=TO Lemma=to NamedEntityTag=O]
[Text=Denver CharacterOffsetBegin=93 CharacterOffsetEnd=99 PartOfSpeech=NNP Lemma=Denver NamedEntityTag=CITY]
[Text=. CharacterOffsetBegin=99 CharacterOffsetEnd=100 PartOfSpeech=. Lemma=. NamedEntityTag=O]

Dependency Parse (enhanced plus plus dependencies):
root(ROOT-0, tokenize-1)
punct(tokenize-1, ,-2)
conj(tokenize-1, ssplit-3)
punct(tokenize-1, ,-4)
conj(tokenize-1, pos-5)
punct(tokenize-1, ,-6)
conj(tokenize-1, lemma-7)
punct(tokenize-1, ,-8)
conj(tokenize-1, ner-9)
punct(tokenize-1, ,-10)
conj(tokenize-1, parse-11)
punct(tokenize-1, ,-12)
conj(tokenize-1, dcoref-13)
punct(tokenize-1, --14)
compound(input.txtList-16, file-15)
dep(tokenize-1, input.txtList-16)
iobj(input.txtList-16, me-17)
det(seats-19, the-18)
dobj(input.txtList-16, seats-19)
case(flight-22, on-20)
det(flight-22, the-21)
nmod:on(seats-19, flight-22)
case(Denver-24, to-23)
nmod:to(flight-22, Denver-24)
punct(tokenize-1, .-25)

Extracted the following NER entity mentions:
Denver CITY

Sentence #1 (8 tokens):
London is the capital of United Kingdom.

Tokens:
[Text=London CharacterOffsetBegin=0 CharacterOffsetEnd=6 PartOfSpeech=NNP Lemma=London NamedEntityTag=CITY]
[Text=is CharacterOffsetBegin=7 CharacterOffsetEnd=9 PartOfSpeech=VBZ Lemma=be NamedEntityTag=O]
[Text=the CharacterOffsetBegin=10 CharacterOffsetEnd=13 PartOfSpeech=DT Lemma=the NamedEntityTag=O]

[Text=capital CharacterOffsetBegin=14 CharacterOffsetEnd=21 PartOfSpeech=NN Lemma=capital NamedEntityTag=O]
[Text=of CharacterOffsetBegin=22 CharacterOffsetEnd=24 PartOfSpeech=IN Lemma=of NamedEntityTag=O]
[Text=United CharacterOffsetBegin=25 CharacterOffsetEnd=31 PartOfSpeech=NNP Lemma=United NamedEntityTag=COUNTRY]
[Text=Kingdom CharacterOffsetBegin=32 CharacterOffsetEnd=39 PartOfSpeech=NNP Lemma=Kingdom NamedEntityTag=COUNTRY]
[Text=. CharacterOffsetBegin=39 CharacterOffsetEnd=40 PartOfSpeech=. Lemma=. NamedEntityTag=O]

Dependency Parse (enhanced plus plus dependencies):
root(ROOT-0, capital-4)
nsubj(capital-4, London-1)
cop(capital-4, is-2)
det(capital-4, the-3)
case(Kingdom-7, of-5)
compound(Kingdom-7, United-6)
nmod:of(capital-4, Kingdom-7)
punct(capital-4, .-8)

Extracted the following NER entity mentions:
London CITY
United Kingdom COUNTRY

Task 6

To process a list of files: the filelist parameter is used. (StanfordCoreNLP, 2018)

```
java -cp "*" -Xmx2g edu.stanford.nlp.pipeline.StanfordCoreNLP [ -props myprops.props ] -filelist
filelist.txt
```

where the -filelist parameter points to a file whose content lists all files to be processed (one per line).

Properties file: A .properties file is a file extension that stores the configurable parameters of an application. For example, with an annotators .property file should contain the list of annotators applicable separated by commas.

Task 7: Adding a .txt file – task7.txt with the given annotators.

- Queen Mary University is located in London. It is a great university.

The default output for the txt file is an xml. So it would be saved as task7.txt.xml.

The xml output file has been attached with the report.

The following code has been implemented:
java -cp "*" -Xmx3g edu.stanford.nlp.pipeline.StanfordCoreNLP -annotators tokenize,ssplit,pos,lemma,ner,parse,dcoref -file task7.txt

Task 8:
Now to generate the output in .conll format for the same file , the following code has been run:

java -cp "*" -Xmx3g edu.stanford.nlp.pipeline.StanfordCoreNLP -annotators tokenize,ssplit,pos,lemma,ner,parse,dcoref -outputFormat conll -file task7.txt

The output has been provided with the report.

Task 9

Instead of adding the annotators in the command line, a .properties file called StanfordCoreNLP.properties has been created that includes the list of annotators as shown below.

```
annotators = tokenize, ssplit, pos, lemma, ner, parse, dcoref
```

Then the following command has been run in the command window. The output file has been saved as a text again. So it is saved as task7.txt.out.

java -cp "*" -Xmx5g edu.stanford.nlp.pipeline.StanfordCoreNLP -props StanfordCoreNLP.properties -file task7.txt -outputFormat text

Task 10

The deterministic coreference resolver is based on a multi-pass sieve since it applies different levels of deterministic coreference models sequentially from highest to lowest precision , instead of applying all the filters simultaneously. Each tier adds on the entity clusters built by preceding models in the sieve, making sure that stronger features are given weightage over weaker features. Linguistic intuition can form the basis of precision . Else, precision can also be worked out from a coreference corpus. Thus the multi-pass sieve generates better results than a single-pass model (Lee et all, 2013).

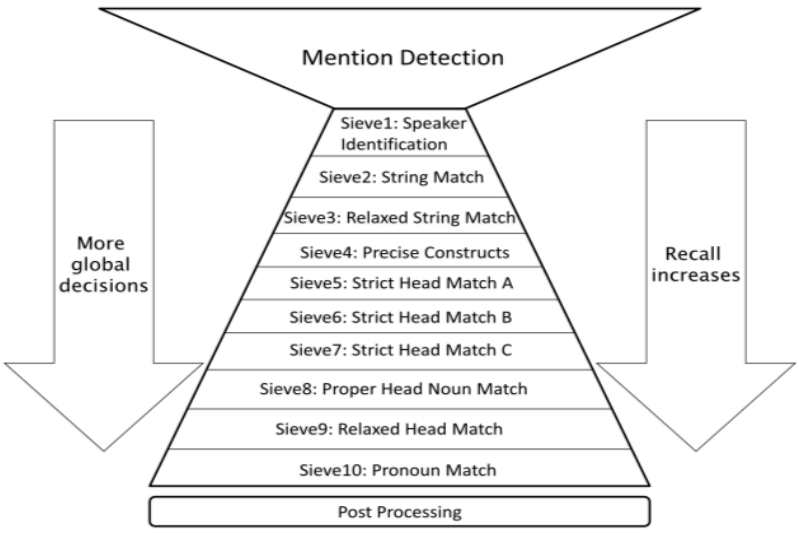


Figure 1
The architecture of our coreference system.

Task 11
The coreference system has been loaded by running the corenlp.sh and it automatically detects the StanfordCoreNLP.properties file that has been created with the annotators.

Coreference Success

Sentence #1 (9 tokens):
Elizabeth said she likes her tea without sugar.

Tokens:
[Text=Elizabeth CharacterOffsetBegin=0 CharacterOffsetEnd=9 PartOfSpeech=NNP Lemma=Elizabeth NamedEntityTag=PERSON]
[Text=said CharacterOffsetBegin=10 CharacterOffsetEnd=14 PartOfSpeech=VBD Lemma=say NamedEntityTag=O]
[Text=she CharacterOffsetBegin=15 CharacterOffsetEnd=18 PartOfSpeech=PRP Lemma=she NamedEntityTag=O]
[Text=likes CharacterOffsetBegin=19 CharacterOffsetEnd=24 PartOfSpeech=VBZ Lemma=like NamedEntityTag=O]
[Text=her CharacterOffsetBegin=25 CharacterOffsetEnd=28 PartOfSpeech=PRP$ Lemma=she NamedEntityTag=O]
[Text=tea CharacterOffsetBegin=29 CharacterOffsetEnd=32 PartOfSpeech=NN Lemma=tea NamedEntityTag=O]
[Text=without CharacterOffsetBegin=33 CharacterOffsetEnd=40 PartOfSpeech=IN Lemma=without NamedEntityTag=O]
[Text=sugar CharacterOffsetBegin=41 CharacterOffsetEnd=46 PartOfSpeech=NN Lemma=sugar NamedEntityTag=O]
[Text=. CharacterOffsetBegin=46 CharacterOffsetEnd=47 PartOfSpeech=. Lemma=. NamedEntityTag=O]

Dependency Parse (enhanced plus plus dependencies):
root(ROOT-0, said-2)
nsubj(said-2, Elizabeth-1)
nsubj(likes-4, she-3)
ccomp(said-2, likes-4)
nmod:poss(tea-6, her-5)
dobj(likes-4, tea-6)
case(sugar-8, without-7)
nmod:without(likes-4, sugar-8)
punct(said-2, .-9)

Extracted the following NER entity mentions:
Elizabeth PERSON
she PERSON
her PERSON

Coreference set:
(1,3,[3,4]) -> (1,1,[1,2]), that is: "she" -> "Elizabeth"
(1,5,[5,6]) -> (1,1,[1,2]), that is: "her" -> "Elizabeth"

As can be seen , it has been coreferenced correctly.

Sentence #1 (19 tokens):
We all support the cause of social welfare as our society is an integral part of our life.

Tokens:
[Text=We CharacterOffsetBegin=0 CharacterOffsetEnd=2 PartOfSpeech=PRP Lemma=we NamedEntityTag=O]
[Text=all CharacterOffsetBegin=3 CharacterOffsetEnd=6 PartOfSpeech=DT Lemma=all NamedEntityTag=O]
[Text=support CharacterOffsetBegin=7 CharacterOffsetEnd=14 PartOfSpeech=VBP Lemma=support NamedEntityTag=O]
[Text=the CharacterOffsetBegin=15 CharacterOffsetEnd=18 PartOfSpeech=DT Lemma=the NamedEntityTag=O]
[Text=cause CharacterOffsetBegin=19 CharacterOffsetEnd=24 PartOfSpeech=NN Lemma=cause NamedEntityTag=O]
[Text=of CharacterOffsetBegin=25 CharacterOffsetEnd=27 PartOfSpeech=IN Lemma=of NamedEntityTag=O]
[Text=social CharacterOffsetBegin=28 CharacterOffsetEnd=34 PartOfSpeech=JJ Lemma=social NamedEntityTag=IDEOLOGY]
[Text=welfare CharacterOffsetBegin=35 CharacterOffsetEnd=42 PartOfSpeech=NN Lemma=welfare NamedEntityTag=IDEOLOGY]
[Text=as CharacterOffsetBegin=43 CharacterOffsetEnd=45 PartOfSpeech=IN Lemma=as NamedEntityTag=O]
[Text=our CharacterOffsetBegin=46 CharacterOffsetEnd=49 PartOfSpeech=PRP$ Lemma=we NamedEntityTag=O]
[Text=society CharacterOffsetBegin=50 CharacterOffsetEnd=57 PartOfSpeech=NN Lemma=society NamedEntityTag=O]
[Text=is CharacterOffsetBegin=58 CharacterOffsetEnd=60 PartOfSpeech=VBZ Lemma=be NamedEntityTag=O]
[Text=an CharacterOffsetBegin=61 CharacterOffsetEnd=63 PartOfSpeech=DT Lemma=a NamedEntityTag=O]
[Text=integral CharacterOffsetBegin=64 CharacterOffsetEnd=72 PartOfSpeech=JJ Lemma=integral NamedEntityTag=O]
[Text=part CharacterOffsetBegin=73 CharacterOffsetEnd=77 PartOfSpeech=NN Lemma=part NamedEntityTag=O]
[Text=of CharacterOffsetBegin=78 CharacterOffsetEnd=80 PartOfSpeech=IN Lemma=of NamedEntityTag=O]
[Text=our CharacterOffsetBegin=81 CharacterOffsetEnd=84 PartOfSpeech=PRP$ Lemma=we NamedEntityTag=O]
[Text=life CharacterOffsetBegin=85 CharacterOffsetEnd=89 PartOfSpeech=NN Lemma=life NamedEntityTag=O]
[Text=. CharacterOffsetBegin=89 CharacterOffsetEnd=90 PartOfSpeech=. Lemma=. NamedEntityTag=O]

Dependency Parse (enhanced plus plus dependencies):
root(ROOT-0, support-3)
nsubj(support-3, We-1)
det(We-1, all-2)
det(cause-5, the-4)
dobj(support-3, cause-5)
case(welfare-8, of-6)
amod(welfare-8, social-7)
nmod:of(cause-5, welfare-8)
mark(part-15, as-9)
nmod:poss(society-11, our-10)
nsubj(part-15, society-11)
cop(part-15, is-12)

det(part-15, an-13)
amod(part-15, integral-14)
advcl:as(support-3, part-15)
case(life-18, of-16)
nmod:poss(life-18, our-17)
nmod:of(part-15, life-18)
punct(support-3, .-19)

Extracted the following NER entity mentions:
social welfare IDEOLOGY

Coreference set:
(1,10,[10,11]) -> (1,1,[1,3]), that is: "our" -> "We all"
(1,17,[17,18]) -> (1,1,[1,3]), that is: "our" -> "We all"

As can be seen, it has been coreferenced correctly.

Coreference Failures:

NLP> Lucy called Anna and asked her if she wanted to go for lunch.

Sentence #1 (14 tokens):
Lucy called Anna and asked her if she wanted to go for lunch.

Tokens:
[Text=Lucy CharacterOffsetBegin=0 CharacterOffsetEnd=4 PartOfSpeech=NNP Lemma=Lucy NamedEntityTag=PERSON]
[Text=called CharacterOffsetBegin=5 CharacterOffsetEnd=11 PartOfSpeech=VBD Lemma=call NamedEntityTag=O]
[Text=Anna CharacterOffsetBegin=12 CharacterOffsetEnd=16 PartOfSpeech=NNP Lemma=Anna NamedEntityTag=PERSON]
[Text=and CharacterOffsetBegin=17 CharacterOffsetEnd=20 PartOfSpeech=CC Lemma=and NamedEntityTag=O]
[Text=asked CharacterOffsetBegin=21 CharacterOffsetEnd=26 PartOfSpeech=VBD Lemma=ask NamedEntityTag=O]
[Text=her CharacterOffsetBegin=27 CharacterOffsetEnd=30 PartOfSpeech=PRP$ Lemma=she NamedEntityTag=O]
[Text=if CharacterOffsetBegin=31 CharacterOffsetEnd=33 PartOfSpeech=IN Lemma=if NamedEntityTag=O]
[Text=she CharacterOffsetBegin=34 CharacterOffsetEnd=37 PartOfSpeech=PRP Lemma=she NamedEntityTag=O]
[Text=wanted CharacterOffsetBegin=38 CharacterOffsetEnd=44 PartOfSpeech=VBD Lemma=want NamedEntityTag=O]
[Text=to CharacterOffsetBegin=45 CharacterOffsetEnd=47 PartOfSpeech=TO Lemma=to NamedEntityTag=O]
[Text=go CharacterOffsetBegin=48 CharacterOffsetEnd=50 PartOfSpeech=VB Lemma=go NamedEntityTag=O]
[Text=for CharacterOffsetBegin=51 CharacterOffsetEnd=54 PartOfSpeech=IN Lemma=for NamedEntityTag=O]
[Text=lunch CharacterOffsetBegin=55 CharacterOffsetEnd=60 PartOfSpeech=NN Lemma=lunch NamedEntityTag=O]
[Text=. CharacterOffsetBegin=60 CharacterOffsetEnd=61 PartOfSpeech=. Lemma=. NamedEntityTag=O]

Dependency Parse (enhanced plus plus dependencies):
root(ROOT-0, called-2)

nsubj(called-2, Lucy-1)
nsubj(asked-5, Lucy-1)
dobj(called-2, Anna-3)
cc(called-2, and-4)
conj:and(called-2, asked-5)
dobj(asked-5, her-6)
mark(wanted-9, if-7)
nsubj(wanted-9, she-8)
nsubj:xsubj(go-11, she-8)
advcl:if(asked-5, wanted-9)
mark(go-11, to-10)
xcomp(wanted-9, go-11)

case(lunch-13, for-12)
nmod:for(go-11, lunch-13)
punct(called-2, .-14)

Extracted the following NER entity mentions:
Lucy PERSON
Anna PERSON
her PERSON
she PERSON

Coreference set:
(1,6,[6,7]) -> (1,1,[1,2]), that is: "her" -> "Lucy"
(1,8,[8,9]) -> (1,1,[1,2]), that is: "she" -> "Lucy"

As can be seen, her and she refers to Anna , not Lucy in this case.

Task 12.

The StanfordCoreNLP.properties file has been modified to include animacy and gender in the following way:

```
# task 9
annotators = tokenize, ssplit, pos, lemma, ner, parse, dcoref

# task 12 - add the below

dcoref.animate
dcoref.inanimate
dcoref.male
dcoref.female
dcoref.neutral
```

After adding the animacy and gender, when tested for the same sentences above in task11, it does not change the results.

References

- Lee, H., Chang, A., Peirsman, Y., Chambers, N., Surdeanu, M. and Jurafsky, D. (2013). Deterministic Coreference Resolution Based on Entity-Centric, Precision-Ranked Rules. *Computational Linguistics*, 39(4), pp.885-916.

- Stanfordnlp.github.io. (2018). *Stanford CoreNLP – Natural language software* | *Stanford CoreNLP*. [online] Available at: https://stanfordnlp.github.io/CoreNLP/ [Accessed 29 Mar. 2018].