Good File

PROJECT 4

PREPARED FOR

Dr. Sabine Bergler

PREPARED BY

Karandeep

40104845

TABLE OF CONTENTS

Introduction:

About NLTK - NLTK aka the Natural Language Toolkit, is a suite of open source Python modules, data sets, and tutorials supporting research and development in Natural Language Processing. It contains text processing libraries for tokenization, parsing, classification, stemming, tagging and semantic reasoning. It also includes graphical demonstrations and sample data sets as well as accompanied by a cook book and a book which explains the principles behind the underlying language processing tasks that NLTK supports.

Purpose:

Hidden information often lies deep within the boundaries of what we can perceive with our eyes and our ears. Some look to data for that purpose, and most of the time, data can tell us more than we thought was imaginable. But sometimes data might not be clear cut enough to perform any sort of analytics. Language, tone, and sentence structure can explain a lot about how people are feeling, and can even be used to predict how people might feel about similar topics using a combination of the Natural Language Toolkit, a Python library used for analyzing text, and machine learning.

For this project I was to develop a Feature grammar to parse given sentences and judge the sentiment behind the sentence. The console has options to run **Sentiment Analyzer** either on a good inbuilt data or bad inbuilt data sentences to parse. Analysis of data has been done in a way that is easily comprehensible. Overall goal of the project was to create a parser and a FCFG which filters data by using NLTK access commands and creates sentence trees and visualize them in a more informative way. NLTK is not perfect and lacks in some areas which is being explored below.

Good Validation Data (Feature Context Free Grammar)/ Sentiment Parser:

Examples:

SMALL SENTENCES

- Good Sentence 1 - It is a compelling Story
  OUTPUT:

```
Sentence ->  it is a compelling story

(S[-INV]
  (NP[-WH] it)
  (VP[]
    (V[+AUX] is)
    (NP[]
      (NP[]
        (NP[-WH] (DT[] a))
        (NP[SNT='POS'] (ADJ[SNT='POS'] compelling)))
      (NP[-WH] story))))

          ************************************
          #     Project 3 Output: The sentence is       Positive
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Positive
          ************************************
```

- GoodSentence 2
  OUTPUT:

```
Sentence ->  it has low impact

(S[-INV]
  (NP[-WH] it)
  (VP[]
    (V[+AUX] has)
    (NP[] (NP[SNT='NEG'] (ADJ[SNT='NEG'] low)) (NP[-WH] impact))))

          ************************************
          #     Project 3 Output: The sentence is       Negative
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Negative
          ************************************
```

- Good Sentence 3
  OUTPUT:

```
Sentence ->  it has low impact but it is a compelling story

(S[-INV]
  (NP[-WH] it)
  (VP[]
    (VP[]
      (V[+AUX] has)
      (NP[] (NP[SNT='NEG'] (ADJ[SNT='NEG'] low)) (NP[-WH] impact)))
    (SBar[]
      (Comp[] but)
      (S[-INV]
        (NP[-WH] it)
        (S[-INV]
          (V[+AUX] is)
          (NP[]
            (NP[]
              (NP[-WH] (DT[] a))
              (NP[SNT='POS'] (ADJ[SNT='POS'] compelling)))
            (NP[-WH] story)))))))

          ****************************************
          #     Project 3 Output: The sentence is       Positive
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Positive
          ****************************************
```

- Good Sentence 4
  OUTPUT:

```
Sentence ->  it is a compelling story , but it has low impact

(S[-INV]
  (NP[-WH] it)
  (VP[]
    (VP[]
      (V[+AUX] is)
      (NP[]
        (NP[]
          (NP[-WH] (DT[] a))
          (NP[SNT='POS'] (ADJ[SNT='POS'] compelling)))
        (NP[-WH] story)))
    (SPR[] ,)
    (SBar[]
      (Comp[] but)
      (S[-INV]
        (NP[-WH] it)
        (VP[]
          (V[+AUX] has)
          (NP[]
            (NP[SNT='NEG'] (ADJ[SNT='NEG'] low))
            (NP[-WH] impact)))))))

          ****************************************
          #     Project 3 Output: The sentence is       Negative
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Negative
          ****************************************
```

- Good Sentence 5
  OUTPUT:

```
Sentence ->  it has gut-wrenching impact and it is a compelling story

(S[-INV]
  (NP[-WH] it)
  (VP[]
    (VP[]
      (V[+AUX] has)
      (NP[]
        (NP[SNT='NEG'] (ADJ[SNT='NEG'] gut-wrenching))
        (NP[-WH] impact)))
    (SBar[]
      (Comp[] and)
      (S[-INV]
        (NP[-WH] it)
        (S[-INV]
          (V[+AUX] is)
          (NP[]
            (NP[]
              (NP[-WH] (DT[] a))
              (NP[SNT='POS'] (ADJ[SNT='POS'] compelling)))
            (NP[-WH] story)))))))

          ****************************************
          #     Project 3 Output: The sentence is       Neutral
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Positive
          ****************************************
```

- Good Sentence 6
  OUTPUT:

```
Sentence ->  this does not have gut-wrenching impact but it is a compelling story

(S[-INV]
  (NP[-WH] (DT[] this))
  (VP[]
    (V[+AUX] (V[+AUX] does) (ADVB[+NEG] not))
    (VP[]
      (VP[]
        (V[+AUX] have)
        (NP[]
          (NP[SNT='NEG'] (ADJ[SNT='NEG'] gut-wrenching))
          (NP[-WH] impact)))
      (SBar[]
        (Comp[] but)
        (S[-INV]
          (NP[-WH] it)
          (S[-INV]
            (V[+AUX] is)
            (NP[]
              (NP[]
                (NP[-WH] (DT[] a))
                (NP[SNT='POS'] (ADJ[SNT='POS'] compelling)))
              (NP[-WH] story)))))))

          ****************************************
          #     Project 3 Output: The sentence is       Positive
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Positive
          ****************************************
```

- Good Sentence 7
  OUTPUT:

```
Sentence ->  this compelling story with gut-wrenching impact

(S[-INV]
  (NP[]
    (NP[]
      (NP[]
        (NP[]
          (NP[-WH] (DT[] this))
          (NP[SNT='POS'] (ADJ[SNT='POS'] compelling)))
        (NP[-WH] story))
      (NP[]
        (PRP[] with)
        (NP[SNT='NEG'] (ADJ[SNT='NEG'] gut-wrenching))))
    (NP[-WH] impact)))

          ***************************************
          #     Project 3 Output: The sentence is       Neutral
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Positive
          ***************************************
```

- Good Sentence 8
  OUTPUT:

```
Sentence ->  a perfect example

(S[-INV]
  (NP[]
    (NP[]
      (NP[-WH] (DT[] a))
      (NP[SNT='POS'] (ADJ[SNT='POS'] perfect)))
    (NP[-WH] example)))

          ***************************************
          #     Project 3 Output: The sentence is       Positive
          #     SSAP Baseline Output: The sentence is   Positive
          #     Project 4 Output: The sentence is       Positive
          ***************************************
```

- Good Sentence 9
  OUTPUT:

```
Sentence ->  manipulative movie making

(S[-INV]
  (NP[]
    (NP[SNT='NEG'] (ADJ[SNT='NEG'] manipulative))
    (NP[-WH] movie))
  (VP[] (V[-AUX, SUBCAT='intrans'] making)))

          ***************************************
          #     Project 3 Output: The sentence is       Negative
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Negative
          ***************************************
```

- Good Sentence 10
  OUTPUT:

```
Sentence ->  shamelessly manipulative movie making

(S[-INV]
  (NP[]
    (NP[]
      (NP[SNT='NEG'] (ADJ[SNT='NEG'] shamelessly))
      (NP[SNT='NEG'] (ADJ[SNT='NEG'] manipulative)))
    (NP[-WH] movie))
  (VP[] (V[-AUX, SUBCAT='intrans'] making)))

         ************************************
         #     Project 3 Output: The sentence is       Negative
         #     SSAP Baseline Output: The sentence is   Neutral
         #     Project 4 Output: The sentence is       Negative
         ************************************
```

- Good Sentence 11
  OUTPUT:

```
Sentence ->  well-intentioned movie making

(S[-INV]
  (NP[]
    (NP[SNT='POS'] (ADJ[SNT='POS'] well-intentioned))
    (NP[-WH] movie))
  (VP[] (V[-AUX, SUBCAT='intrans'] making)))

          ************************************
          #     Project 3 Output: The sentence is       Positive
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Positive
          ************************************
```

- Good Sentence 12
  OUTPUT:

```
Sentence ->  rancid movie making

(S[-INV]
  (NP[] (NP[SNT='NEG'] (ADJ[SNT='NEG'] rancid)) (NP[-WH] movie))
  (VP[] (V[-AUX, SUBCAT='intrans'] making)))

          ************************************
          #     Project 3 Output: The sentence is       Negative
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Negative
          ************************************
```

- Good Sentence 13
  OUTPUT:

```
Sentence ->  well-intentioned but manipulative movie making

(S[-INV]
  (NP[]
    (NP[]
      (NP[]
        (NP[SNT='POS'] (ADJ[SNT='POS'] well-intentioned))
        (Comp[] but))
      (NP[SNT='NEG'] (ADJ[SNT='NEG'] manipulative)))
    (NP[-WH] movie))
  (VP[] (V[-AUX, SUBCAT='intrans'] making)))

          ************************************
          #     Project 3 Output: The sentence is       Negative
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Negative
          ************************************
```

- Good Sentence 14
  OUTPUT:

```
Sentence ->  a perfect example of well-intentioned but manipulative movie making

(S[-INV]
  (NP[]
    (NP[]
      (NP[]
        (NP[]
          (NP[-WH] (DT[] a))
          (NP[SNT='POS'] (ADJ[SNT='POS'] perfect)))
        (NP[]
          (NP[]
            (NP[-WH] example)
            (NP[]
              (PRP[] of)
              (NP[SNT='POS'] (ADJ[SNT='POS'] well-intentioned))))
          (Comp[] but)))
      (NP[SNT='NEG'] (ADJ[SNT='NEG'] manipulative)))
    (NP[-WH] movie))
  (VP[] (V[-AUX, SUBCAT='intrans'] making)))

          ************************************
          #     Project 3 Output: The sentence is       Negative
          #     SSAP Baseline Output: The sentence is   Positive
          #     Project 4 Output: The sentence is       Negative
          ************************************
```

NEGATION SENTENCES:

- Negation Sentence 1
  OUTPUT:

```
Sentence ->  it is not a bad movie

(S[-INV]
  (NP[-WH] it)
  (VP[]
    (V[+AUX] (V[+AUX] is) (ADVB[+NEG] not))
    (NP[]
      (NP[] (NP[-WH] (DT[] a)) (NP[SNT='NEG'] (ADJ[SNT='NEG'] bad)))
      (NP[-WH] movie))))

          ************************************
          #     Project 3 Output: The sentence is       Negative
          #     SSAP Baseline Output: The sentence is   Negative
          #     Project 4 Output: The sentence is       Positive
          ************************************
```

- Negation Sentence 2
  OUTPUT:

```
Sentence ->  it is pretty disgusting

(S[-INV]
  (NP[-WH] it)
  (VP[]
    (V[+AUX] is)
    (NP[]
      (NP[SNT='POS'] (ADJ[SNT='POS'] pretty))
      (NP[SNT='NEG'] (ADJ[SNT='NEG'] disgusting)))))

          ************************************
          #     Project 3 Output: The sentence is       Neutral
          #     SSAP Baseline Output: The sentence is   Negative
          #     Project 4 Output: The sentence is       Negative
          ************************************
```

- Negation Sentence 3
  OUTPUT:

```
Sentence ->  it is not a good movie

(S[-INV]
  (NP[-WH] it)
  (VP[]
    (V[+AUX] (V[+AUX] is) (ADVB[+NEG] not))
    (NP[]
      (NP[] (NP[-WH] (DT[] a)) (NP[SNT='POS'] (ADJ[SNT='POS'] good)))
      (NP[-WH] movie))))

          ***************************************
          #     Project 3 Output: The sentence is       Positive
          #     SSAP Baseline Output: The sentence is   Positive
          #     Project 4 Output: The sentence is       Negative
          ***************************************
```

- Negation Sentence 4
  OUTPUT:

```
Sentence ->  this does not have compelling factor in it

(S[-INV]
  (NP[-WH] (DT[] this))
  (VP[]
    (V[+AUX] (V[+AUX] does) (ADVB[+NEG] not) (V[+AUX] have))
    (NP[]
      (NP[]
        (NP[SNT='POS'] (ADJ[SNT='POS'] compelling))
        (NP[-WH] factor))
      (NP[] (PRP[] in) (NP[-WH] it)))))

          ***************************************
          #     Project 3 Output: The sentence is       Positive
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Negative
          ***************************************
```

- Negation Sentence 5
  OUTPUT:

```
Sentence ->  this movie is amazingly awful

(S[-INV]
  (NP[] (NP[-WH] (DT[] this)) (NP[-WH] movie))
  (VP[]
    (V[+AUX] is)
    (NP[]
      (NP[SNT='POS'] (ADJ[SNT='POS'] amazingly))
      (NP[SNT='NEG'] (ADJ[SNT='NEG'] awful)))))

          ***************************************
          #     Project 3 Output: The sentence is       Neutral
          #     SSAP Baseline Output: The sentence is   Negative
          #     Project 4 Output: The sentence is       Negative
          ***************************************
```

- Negation Sentence 6
  OUTPUT:

```
Sentence ->  it is neither bad nor good movie

(S[-INV]
  (NP[-WH] it)
  (VP[]
    (VP[]
      (V[+AUX] is)
      (NP[]
        (NP[-WH] (DT[] neither))
        (NP[SNT='NEG'] (ADJ[SNT='NEG'] bad))))
    (SBar[]
      (Comp[] nor)
      (S[-INV]
        (NP[SNT='POS'] (ADJ[SNT='POS'] good))
        (S[-INV] (NP[-WH] movie))))))

          ************************************
          #     Project 3 Output: The sentence is       Neutral
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Neutral
          ************************************
```

BIG COMPLEX SENTENCES:

- Big Complex Sentence 1
  OUTPUT:

```
Please enter your choice: 2
this may not have the dramatic gut-wrenching impact of other holocaust films but it is a compelling story  mainly because of the way it i
s told by the people who were there

(S[-INV]
  (NP[-WH] (DT[] this))
  (VP[]
    (V[+AUX] (V[+AUX] may) (ADVB[+NEG] not))
    (VP[]
      (VP[]
        (V[+AUX] have)
        (NP[]
          (NP[]
            (NP[]
              (NP[]
                (NP[]
                  (NP[]
                    (NP[]
                      (NP[-WH] (DT[] the))
                      (NP[SNT='POS'] (ADJ[SNT='POS'] dramatic)))
                    (NP[SNT='NEG'] (ADJ[SNT='NEG'] gut-wrenching)))
                  (NP[-WH] impact))
                (NP[]
                  (PRP[] of)
                  (NP[SNT='NEG'] (ADJ[SNT='NEG'] other))))
              (NP[-WH] holocaust))
            (NP[-WH] films)))
        (SBar[]
          (Comp[] but)
          (S[-INV]
            (NP[-WH] it)
            (S[-INV]
              (V[+AUX] is)
              (NP[]
                (NP[]
                  (NP[-WH] (DT[] a))
                  (NP[SNT='POS'] (ADJ[SNT='POS'] compelling)))
                (NP[-WH] story))))))))
```

```
(S[-INV]
  (NP[]
    (NP[]
      (NP[]
        (NP[] (ADVB[-NEG] mainly))
        (NP[] (PRP[] because) (NP[] (PRP[] of) (NP[-WH] (DT[] the)))))
      (NP[-WH] way))
    (NP[-WH] it))
  (VP[]
    (VP[]
      (V[+AUX] is)
      (VP[]
        (V[+AUX] told)
        (NP[]
          (PRP[] by)
          (NP[]
            (NP[] (NP[-WH] (DT[] the)) (NP[-WH] people))
            (NP[+WH] who)))))
    (SBar[] (V[+AUX] (V[+AUX] were) (ADVB[-NEG] there)))))

          ***************************************
          #     Project 3 Output: The sentence is       Negative
          #     SSAP Baseline Output: The sentence is   Neutral
          #     Project 4 Output: The sentence is       Positive
          ***************************************
```

- Big Complex Sentence 2
  OUTPUT:

```
a perfect example of rancid  well-intentioned but shamelessly manipulative movie making

(S[-INV]
  (NP[-WH] (DT[] a))
  (S[-INV]
    (NP[]
      (NP[]
        (NP[SNT='POS'] (ADJ[SNT='POS'] perfect))
        (NP[-WH] example))
      (NP[] (PRP[] of) (NP[SNT='NEG'] (ADJ[SNT='NEG'] rancid))))))
(S[-INV]
  (NP[]
    (NP[]
      (NP[]
        (NP[]
          (NP[SNT='POS'] (ADJ[SNT='POS'] well-intentioned))
          (Comp[] but))
        (NP[SNT='NEG'] (ADJ[SNT='NEG'] shamelessly)))
      (NP[SNT='NEG'] (ADJ[SNT='NEG'] manipulative)))
    (NP[-WH] movie))
  (VP[] (V[-AUX, SUBCAT='intrans'] making)))

          ***************************************
          #     Project 3 Output: The sentence is       Neutral
          #     SSAP Baseline Output: The sentence is   Positive
          #     Project 4 Output: The sentence is       Negative
          ***************************************
```

PARAGRAPH SENTENCES:

- Paragraph 1
  OUTPUT:

```
this is one of the best book by Crichton. the characters of Karen Ross , Peter Elliot , Munro and Amy are beautifully developed and their
 interactions are exciting , that get lost in the film. this may be the absolute worst disparity in quality between novel and the screen
adaptation. the book is really good. the movie is just dreadful.
(S[-INV]
  (NP[-WH] (DT[] this))
  (VP[]
    (V[+AUX] is)
    (NP[]
      (NP[]
        (NP[]
          (NP[] (NP[] (CD[] one) (PRP[] of)) (NP[-WH] (DT[] the)))
          (NP[SNT='POS'] (ADJ[SNT='POS'] best)))
        (NP[-WH] book))
      (NP[] (PRP[] by) (NP[-WH] Crichton)))))  (S[-INV]
  (S[-INV]
    (NP[] (NP[-WH] (DT[] the)) (NP[-WH] characters))
    (S[-INV]
      (S[-INV]
        (NP[] (PRP[] of) (NP[-WH] Karen))
        (S[-INV] (NP[-WH] Ross)))
      (SPR[] ,)
      (S[-INV] (NP[-WH] Peter) (S[-INV] (NP[-WH] Elliot)))))
  (SPR[] ,)
  (S[-INV]
    (NP[] (NP[] (NP[-WH] Munro) (Comp[] and)) (NP[-WH] Amy))
    (VP[]
      (VP[]
        (VP[]
          (V[+AUX] are)
          (NP[SNT='POS'] (ADJ[SNT='POS'] beautifully)))
        (SBar[] (V[+AUX] developed)))
      (SBar[]
        (Comp[] and)
        (S[-INV]
          (NP[] (PRP[] their) (NP[-WH] interactions))
          (VP[]
            (V[+AUX] are)
            (NP[SNT='POS'] (ADJ[SNT='POS'] exciting)))))))
```

```
(SPR[] ,)
(SBar[]
  (Comp[] that)
  (S[-INV]
    (V[+AUX] get)
    (NP[]
      (NP[]
        (NP[SNT='NEG'] (ADJ[SNT='NEG'] lost))
        (NP[] (PRP[] in) (NP[-WH] (DT[] the))))
      (NP[-WH] film)))))  (S[-INV]
(NP[-WH] (DT[] this))
(VP[]
  (V[+AUX] may)
  (VP[]
    (V[+AUX] be)
    (NP[]
      (NP[]
        (NP[]
          (NP[]
            (NP[]
              (NP[]
                (NP[]
                  (NP[]
                    (NP[-WH] (DT[] the))
                    (NP[] (ADVB[-NEG] absolute)))
                  (NP[SNT='NEG'] (ADJ[SNT='NEG'] worst)))
                (NP[-WH] disparity))
              (NP[]
                (PRP[] in)
                (NP[]
                  (NP[]
                    (NP[-WH] quality)
                    (NP[] (PRP[] between) (NP[-WH] novel)))
                  (Comp[] and))))
            (NP[-WH] (DT[] the)))
          (NP[-WH] screen))
        (NP[-WH] adaptation)))))  (S[-INV]
```

```
(NP[] (NP[-WH] (DT[] the)) (NP[-WH] book))
(VP[]
  (V[+AUX] (V[+AUX] is) (ADVB[-NEG] really))
  (NP[SNT='POS'] (ADJ[SNT='POS'] good))))  (S[-INV]
(NP[] (NP[-WH] (DT[] the)) (NP[-WH] movie))
(VP[]
  (V[+AUX] (V[+AUX] is) (ADVB[-NEG] just))
  (NP[SNT='NEG'] (ADJ[SNT='NEG'] dreadful))))

      ************************************
      #     Project 3 Output: The sentence is      Neutral
      #     SSAP Baseline Output: The sentence is  Negative
      #     Project 4 Output: The sentence is      Positive
      ************************************
```

- Paragraph 2
  OUTPUT:

```
there is no movie I have been more prepared to dislike than this one. How dare some Aussie come over here and tell us about the meaning o
f one of the great works of American literature. Especially this Aussie , Baz Luhrmann , who is known to overload.
(S[-INV]
  (NP[] (ADVB[-NEG] there))
  (S[-INV]
    (V[+AUX] is)
    (NP[] (NP[] (NP[-WH] (DT[] no)) (NP[-WH] movie)) (NP[-WH] I))
    (VP[]
      (V[+AUX] have)
      (VP[]
        (V[+AUX] (V[+AUX] been) (ADVB[-NEG] more))
        (NP[]
          (NP[]
            (NP[SNT='POS'] (ADJ[SNT='POS'] prepared))
            (NP[]
              (PRP[] to)
              (NP[SNT='NEG'] (ADJ[SNT='NEG'] dislike))))
          (NP[] (PRP[] than) (NP[-WH] (DT[] this) (CD[] one))))))))  (S[-INV]
  (NP[] (ADVB[+NEG] How))
  (S[-INV]
    (V[+AUX] dare)
    (NP[] (NP[-WH] (DT[] some)) (NP[-WH] Aussie))
    (VP[]
      (VP[]
        (V[+AUX] come)
        (NP[] (PRP[] over) (NP[] (ADVB[+NEG] here))))
      (SBar[]
        (Comp[] and)
        (S[-INV]
          (V[+AUX] tell)
          (NP[]
```

```
                    (NP[]
                      (NP[-WH] us)
                      (NP[] (PRP[] about) (NP[-WH] (DT[] the))))
                    (NP[-WH] meaning))
                  (NP[] (PRP[] of) (NP[] (CD[] one) (PRP[] of))))
                (NP[-WH] (DT[] the)))
              (NP[SNT='POS'] (ADJ[SNT='POS'] great)))
            (NP[SNT='POS'] (ADJ[SNT='POS'] works)))
          (NP[] (PRP[] of) (NP[-WH] American)))
        (NP[-WH] literature)))))))  (S[-INV]
  (S[-INV]
    (NP[] (ADVB[-NEG] Especially))
    (S[-INV]
      (S[-INV] (NP[-WH] (DT[] this)) (S[-INV] (NP[-WH] Aussie)))
      (SPR[] ,)
      (S[-INV] (NP[-WH] Baz) (S[-INV] (NP[-WH] Luhrmann)))))
  (SPR[] ,)
  (S[-INV]
    (NP[+WH] who)
    (VP[]
      (V[+AUX] is)
      (VP[] (V[+AUX] known) (NP[] (PRP[] to) (NP[-WH] overload))))))

        ************************************
        #     Project 3 Output: The sentence is       Neutral
        #     SSAP Baseline Output: The sentence is   Neutral
        #     Project 4 Output: The sentence is       Negative
        ************************************
```

END