```
import nltk
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')
nltk.download("maxent_ne_chunker")
nltk.download('words')
```

```
#output------->
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /root/nltk_data...
[nltk_data]   Unzipping taggers/averaged_perceptron_tagger.zip.
[nltk_data] Downloading package maxent_ne_chunker to
[nltk_data]     /root/nltk_data...
[nltk_data]   Unzipping chunkers/maxent_ne_chunker.zip.
[nltk_data] Downloading package words to /root/nltk_data...
[nltk_data]   Unzipping corpora/words.zip.
True
```

```
#code------->

sentence="WASHINGTON -- In the wake of a string of abuses by New York police officers in the 1990s, Loretta E. Lynch, the top federal pr
print("the pos tagging")
for sent in nltk.sent_tokenize(sentence):
    tagged_tokens=nltk.pos_tag(nltk.word_tokenize(sent))
    print(tagged_tokens)
    for chunk in nltk.ne_chunk(tagged_tokens):
        if hasattr(chunk,"label"):
            print(chunk.label()," ".join(c[0] for c in chunk))
```

```
#optout
```

```
the pos tagging
[('WASHINGTON', 'NNP'), ('--', ':'), ('In', 'IN'), ('the', 'DT'), ('wake', 'NN'), ('of', 'IN'), ('a', 'DT'), ('string', 'NN'), ('of
GPE WASHINGTON
GPE New York
PERSON Loretta E. Lynch
GPE Brooklyn
```

```
#code---->

import nltk
inputfile='sample_ontonotes_final.txt'
outputfile="output.txt"
with open(inputfile,'r',encoding='utf-8') as f_in,open(outputfile,'w',encoding='utf-8')as f_out:
    for line in f_in:
        tokens=nltk.word_tokenize(line.strip())
        tagged=nltk.pos_tag(tokens)
        chunked=nltk.ne_chunk(tagged)
        for chunk in chunked:
            if hasattr(chunk,'label'):
                f_out.write(chunk.label()+' '+' '.join(c[0] for c in chunk))


#output is ---> creates output.txt and write the text in the file
```

Start coding or generate with AI.