import nltk

with open('sample.txt', 'r') as f:

sample = f.read()

sentences = nltk.sent\_tokenize(sample)

tokenized\_sentences = [nltk.word\_tokenize(sentence) for sentence in sentences]

tagged\_sentences = [nltk.pos\_tag(sentence) for sentence in tokenized\_sentences]

chunked\_sentences = nltk.batch\_ne\_chunk(tagged\_sentences, binary=True)

def extract\_entity\_names(t):

entity\_names = []

if hasattr(t, 'node') and t.node:

if t.node == 'NE':

entity\_names.append(' '.join([child[0] for child in t]))

else:

for child in t:

entity\_names.extend(extract\_entity\_names(child))

return entity\_names

entity\_names = []

for tree in chunked\_sentences:

# Print results per sentence

# print extract\_entity\_names(tree)

entity\_names.extend(extract\_entity\_names(tree))

# Print all entity names

#print entity\_names

# Print unique entity names

print set(entity\_names)