app

October 8, 2019

POS Tagging with HMM and Sentence Generation

The training dataset is a subset of the Brown corpus, where each file contains sentences in the form of tokenized words followed by POS tags. Each line contains one sentence. Training dataset can be downloaded from here: https://bit.ly/2kJI0yc The test dataset (which is another subset of the Brown corpus, containing tokenized words but no tags) can be downloaded from here: https://bit.ly/2lMybzP Information regarding the categories of the dataset can be found at: https://bit.ly/2mhF6RT.

Your task is to implement a part-of-speech tagger using a bi-gram HMM. Given an observation sequence of n words wn1, choose the most probable sequence of POS tags tn1. For the questions-below, please submit both code and output.

[Note: During training, for a word to be counted as unknown, the frequency of the word in training set should not exceed a threshold (e.g. 5). You can pick a threshold based on your algorithm design. Also, you can implement smoothing technique based on your own choice, e.g. add-.]

```
In [1]: import glob
        import re
        import math
        import random
        import pandas as pd
        from collections import Counter
        from nltk.util import ngrams
        from nltk.tokenize import sent_tokenize, word_tokenize
```

```
In [2]: def load_data(input_directory):
            word_tokens = []
            tag_tokens = []
            word_tag_tokens = []
            transition_tag_tokens = []
            sentence_count = 0
            for file_name in glob.glob(input_directory + "/*"):
                print("Prepocessing: {}".format(file_name))
                file_pointer = open(file_name, "r")

                for line in file_pointer:

                    # Remove duplicate spaces
```

```
                    file_line_content = re.sub(' +', ' ', line)

                    # Remove new line characters
                    file_line_content = line.replace("\n", " ")

                    # Strip of begin and end spaces
                    file_line_content = file_line_content.strip()

                    # If line is not empty
                    if file_line_content != "":
                        sentence_count = sentence_count + 1
                        line_content_list = file_line_content.split(" ")

                        # Append start tag
                        transition_tag_tokens.append('START')
#                         print(line_content_list)
                        for i in line_content_list:
                            word_tag_tokens.append(i)
                            split_tokens = i.split('/')
                            word = split_tokens[0]
                            tag = split_tokens[-1]
                            word_tokens.append(word)
                            tag_tokens.append(tag)
                            transition_tag_tokens.append(tag)

                        # Append end tag
                        transition_tag_tokens.append('END')

                file_pointer.close()

            return sentence_count, word_tokens, tag_tokens, word_tag_tokens, transition_tag_tok

In [3]: def replace_low_count_words(word_tokens, cut_off_count, word_tag_tokens):
            word_tokens_with_count = Counter(word_tokens)
            candidate_words = {}
            for word in word_tokens_with_count:
                if word_tokens_with_count[word] <= cut_off_count:
                    candidate_words[word] = 1

            ## Remove all the words <= cut_off_count
            for i in range(len(word_tokens)):
                if word_tokens[i] in candidate_words:
                    word_tokens[i] = 'UNK'
                    split_tokens = word_tag_tokens[i].split('/')
                    word = split_tokens[0]
                    tag = split_tokens[-1]
                    word_tag_tokens[i] = 'UNK' + '/' + tag
```

```
            return word_tokens, word_tag_tokens
```

```
In [4]: print("##### Loading train date")
        # train_sentence_count, train_word_tokens, train_tag_tokens, train_word_tag_tokens, \
        #     train_transition_tag_tokens = load_data('../input/custom_train')

        train_sentence_count, train_word_tokens, train_tag_tokens, train_word_tag_tokens, \
            train_transition_tag_tokens = load_data('../input/brown_train')

        print(" Number of Sentences: {}, Word list count: {}, Tag list count: {}, Transition Ta
              " Word Tag List"
               .format(train_sentence_count, len(train_word_tokens), len(train_tag_tokens), len
              )

        print("### Replace word tokens <= 5 with 'UNK")
        train_word_tokens, train_word_tag_tokens = replace_low_count_words(train_word_tokens,
        print("Number of tokens after replacement: Word - {}, Word_Tag - {}".format(len(train_w
```

```
##### Loading train date
Prepocessing: ../input/brown_train/cd05
Prepocessing: ../input/brown_train/cf37
Prepocessing: ../input/brown_train/cf08
Prepocessing: ../input/brown_train/cl06
Prepocessing: ../input/brown_train/cj68
Prepocessing: ../input/brown_train/cf30
Prepocessing: ../input/brown_train/cd02
Prepocessing: ../input/brown_train/cl01
Prepocessing: ../input/brown_train/cj57
Prepocessing: ../input/brown_train/cf06
Prepocessing: ../input/brown_train/cl08
Prepocessing: ../input/brown_train/cj61
Prepocessing: ../input/brown_train/cf39
Prepocessing: ../input/brown_train/cn05
Prepocessing: ../input/brown_train/cj59
Prepocessing: ../input/brown_train/cf01
Prepocessing: ../input/brown_train/cn02
Prepocessing: ../input/brown_train/cj66
Prepocessing: ../input/brown_train/cj32
Prepocessing: ../input/brown_train/ch07
Prepocessing: ../input/brown_train/cj35
Prepocessing: ../input/brown_train/cb09
Prepocessing: ../input/brown_train/cj03
Prepocessing: ../input/brown_train/cj04
Prepocessing: ../input/brown_train/ch09
Prepocessing: ../input/brown_train/cb07
Prepocessing: ../input/brown_train/cj58
Prepocessing: ../input/brown_train/cn03
Prepocessing: ../input/brown_train/cj67
```

```
Prepocessing: ../input/brown_train/cl09
Prepocessing: ../input/brown_train/cf07
Prepocessing: ../input/brown_train/cj60
Prepocessing: ../input/brown_train/cn04
Prepocessing: ../input/brown_train/cf38
Prepocessing: ../input/brown_train/cj69
Prepocessing: ../input/brown_train/cd03
Prepocessing: ../input/brown_train/cf31
Prepocessing: ../input/brown_train/cj56
Prepocessing: ../input/brown_train/cf36
Prepocessing: ../input/brown_train/cd04
Prepocessing: ../input/brown_train/cj51
Prepocessing: ../input/brown_train/cl07
Prepocessing: ../input/brown_train/cf09
Prepocessing: ../input/brown_train/cj05
Prepocessing: ../input/brown_train/cb06
Prepocessing: ../input/brown_train/cj02
Prepocessing: ../input/brown_train/ch30
Prepocessing: ../input/brown_train/cb01
Prepocessing: ../input/brown_train/cb08
Prepocessing: ../input/brown_train/cj34
Prepocessing: ../input/brown_train/ch06
Prepocessing: ../input/brown_train/ch01
Prepocessing: ../input/brown_train/cj33
Prepocessing: ../input/brown_train/cr03
Prepocessing: ../input/brown_train/ce14
Prepocessing: ../input/brown_train/cg26
Prepocessing: ../input/brown_train/cg19
Prepocessing: ../input/brown_train/cg21
Prepocessing: ../input/brown_train/ce13
Prepocessing: ../input/brown_train/cg17
Prepocessing: ../input/brown_train/ce25
Prepocessing: ../input/brown_train/ca41
Prepocessing: ../input/brown_train/cg28
Prepocessing: ../input/brown_train/cp07
Prepocessing: ../input/brown_train/ce22
Prepocessing: ../input/brown_train/cg10
Prepocessing: ../input/brown_train/ck23
Prepocessing: ../input/brown_train/ca12
Prepocessing: ../input/brown_train/cg44
Prepocessing: ../input/brown_train/ck24
Prepocessing: ../input/brown_train/cg43
Prepocessing: ../input/brown_train/ca15
Prepocessing: ../input/brown_train/ck12
Prepocessing: ../input/brown_train/cc11
Prepocessing: ../input/brown_train/ca23
Prepocessing: ../input/brown_train/cg75
Prepocessing: ../input/brown_train/cg72
```

```
Prepocessing: ../input/brown_train/ca24
Prepocessing: ../input/brown_train/cc16
Prepocessing: ../input/brown_train/cg11
Prepocessing: ../input/brown_train/ce23
Prepocessing: ../input/brown_train/cp06
Prepocessing: ../input/brown_train/cp01
Prepocessing: ../input/brown_train/ce24
Prepocessing: ../input/brown_train/cg16
Prepocessing: ../input/brown_train/ca40
Prepocessing: ../input/brown_train/cg29
Prepocessing: ../input/brown_train/cr05
Prepocessing: ../input/brown_train/ce12
Prepocessing: ../input/brown_train/cg20
Prepocessing: ../input/brown_train/cp08
Prepocessing: ../input/brown_train/cg27
Prepocessing: ../input/brown_train/ce15
Prepocessing: ../input/brown_train/cr02
Prepocessing: ../input/brown_train/cg18
Prepocessing: ../input/brown_train/ck14
Prepocessing: ../input/brown_train/cg73
Prepocessing: ../input/brown_train/cc17
Prepocessing: ../input/brown_train/ca25
Prepocessing: ../input/brown_train/ck13
Prepocessing: ../input/brown_train/ca22
Prepocessing: ../input/brown_train/cc10
Prepocessing: ../input/brown_train/cg74
Prepocessing: ../input/brown_train/ck25
Prepocessing: ../input/brown_train/cg42
Prepocessing: ../input/brown_train/ca14
Prepocessing: ../input/brown_train/ck22
Prepocessing: ../input/brown_train/ca13
Prepocessing: ../input/brown_train/cg45
Prepocessing: ../input/brown_train/cg67
Prepocessing: ../input/brown_train/ca31
Prepocessing: ../input/brown_train/cc03
Prepocessing: ../input/brown_train/cg58
Prepocessing: ../input/brown_train/cc04
Prepocessing: ../input/brown_train/ca36
Prepocessing: ../input/brown_train/cg60
Prepocessing: ../input/brown_train/ca09
Prepocessing: ../input/brown_train/ck07
Prepocessing: ../input/brown_train/cg56
Prepocessing: ../input/brown_train/cg69
Prepocessing: ../input/brown_train/ca07
Prepocessing: ../input/brown_train/ck09
Prepocessing: ../input/brown_train/cg51
Prepocessing: ../input/brown_train/ca38
Prepocessing: ../input/brown_train/ce08
```

```
Prepocessing: ../input/brown_train/cg05
Prepocessing: ../input/brown_train/cp12
Prepocessing: ../input/brown_train/cp15
Prepocessing: ../input/brown_train/cg02
Prepocessing: ../input/brown_train/ce30
Prepocessing: ../input/brown_train/cm05
Prepocessing: ../input/brown_train/cp23
Prepocessing: ../input/brown_train/cg34
Prepocessing: ../input/brown_train/ce06
Prepocessing: ../input/brown_train/cm02
Prepocessing: ../input/brown_train/ce01
Prepocessing: ../input/brown_train/cg33
Prepocessing: ../input/brown_train/cp24
Prepocessing: ../input/brown_train/ck08
Prepocessing: ../input/brown_train/ca06
Prepocessing: ../input/brown_train/cg50
Prepocessing: ../input/brown_train/ca39
Prepocessing: ../input/brown_train/ca01
Prepocessing: ../input/brown_train/cg68
Prepocessing: ../input/brown_train/ca37
Prepocessing: ../input/brown_train/cc05
Prepocessing: ../input/brown_train/cg61
Prepocessing: ../input/brown_train/ck06
Prepocessing: ../input/brown_train/ca08
Prepocessing: ../input/brown_train/cg66
Prepocessing: ../input/brown_train/cc02
Prepocessing: ../input/brown_train/ca30
Prepocessing: ../input/brown_train/ck01
Prepocessing: ../input/brown_train/cg59
Prepocessing: ../input/brown_train/cp25
Prepocessing: ../input/brown_train/cg32
Prepocessing: ../input/brown_train/cm04
Prepocessing: ../input/brown_train/ce07
Prepocessing: ../input/brown_train/cg35
Prepocessing: ../input/brown_train/cp22
Prepocessing: ../input/brown_train/ce31
Prepocessing: ../input/brown_train/cg03
Prepocessing: ../input/brown_train/cp14
Prepocessing: ../input/brown_train/ce09
Prepocessing: ../input/brown_train/cp13
Prepocessing: ../input/brown_train/cg04
Prepocessing: ../input/brown_train/ce36
Prepocessing: ../input/brown_train/cb12
Prepocessing: ../input/brown_train/cj11
Prepocessing: ../input/brown_train/ch23
Prepocessing: ../input/brown_train/cb15
Prepocessing: ../input/brown_train/cj29
Prepocessing: ../input/brown_train/ch24
```

```
Prepocessing: ../input/brown_train/cj16
Prepocessing: ../input/brown_train/cf47
Prepocessing: ../input/brown_train/cb23
Prepocessing: ../input/brown_train/ch12
Prepocessing: ../input/brown_train/cj20
Prepocessing: ../input/brown_train/cb24
Prepocessing: ../input/brown_train/cj18
Prepocessing: ../input/brown_train/cf40
Prepocessing: ../input/brown_train/cj27
Prepocessing: ../input/brown_train/ch15
Prepocessing: ../input/brown_train/cj73
Prepocessing: ../input/brown_train/cn28
Prepocessing: ../input/brown_train/cj80
Prepocessing: ../input/brown_train/cj74
Prepocessing: ../input/brown_train/cl22
Prepocessing: ../input/brown_train/cn10
Prepocessing: ../input/brown_train/cf13
Prepocessing: ../input/brown_train/cl14
Prepocessing: ../input/brown_train/cn26
Prepocessing: ../input/brown_train/cj42
Prepocessing: ../input/brown_train/cn19
Prepocessing: ../input/brown_train/cf25
Prepocessing: ../input/brown_train/cd17
Prepocessing: ../input/brown_train/cj45
Prepocessing: ../input/brown_train/cn21
Prepocessing: ../input/brown_train/cl13
Prepocessing: ../input/brown_train/cd10
Prepocessing: ../input/brown_train/cf22
Prepocessing: ../input/brown_train/cj19
Prepocessing: ../input/brown_train/cf41
Prepocessing: ../input/brown_train/ch14
Prepocessing: ../input/brown_train/cj26
Prepocessing: ../input/brown_train/cf46
Prepocessing: ../input/brown_train/cb22
Prepocessing: ../input/brown_train/cj21
Prepocessing: ../input/brown_train/ch13
Prepocessing: ../input/brown_train/cj28
Prepocessing: ../input/brown_train/cb14
Prepocessing: ../input/brown_train/cj17
Prepocessing: ../input/brown_train/ch25
Prepocessing: ../input/brown_train/cb13
Prepocessing: ../input/brown_train/ch22
Prepocessing: ../input/brown_train/cj10
Prepocessing: ../input/brown_train/cf48
Prepocessing: ../input/brown_train/cj44
Prepocessing: ../input/brown_train/cl12
Prepocessing: ../input/brown_train/cn20
Prepocessing: ../input/brown_train/cf23
```

```
Prepocessing: ../input/brown_train/cd11
Prepocessing: ../input/brown_train/cn27
Prepocessing: ../input/brown_train/cl15
Prepocessing: ../input/brown_train/cj43
Prepocessing: ../input/brown_train/cd16
Prepocessing: ../input/brown_train/cf24
Prepocessing: ../input/brown_train/cn18
Prepocessing: ../input/brown_train/cj75
Prepocessing: ../input/brown_train/cn11
Prepocessing: ../input/brown_train/cl23
Prepocessing: ../input/brown_train/cf12
Prepocessing: ../input/brown_train/cl24
Prepocessing: ../input/brown_train/cn16
Prepocessing: ../input/brown_train/cj72
Prepocessing: ../input/brown_train/cf15
Prepocessing: ../input/brown_train/cn29
Prepocessing: ../input/brown_train/cj36
Prepocessing: ../input/brown_train/ch04
Prepocessing: ../input/brown_train/cj09
Prepocessing: ../input/brown_train/ch03
Prepocessing: ../input/brown_train/cj31
Prepocessing: ../input/brown_train/cj07
Prepocessing: ../input/brown_train/cj38
Prepocessing: ../input/brown_train/cb04
Prepocessing: ../input/brown_train/cb03
Prepocessing: ../input/brown_train/cd01
Prepocessing: ../input/brown_train/cf33
Prepocessing: ../input/brown_train/cl02
Prepocessing: ../input/brown_train/cj54
Prepocessing: ../input/brown_train/cf34
Prepocessing: ../input/brown_train/cn08
Prepocessing: ../input/brown_train/cd06
Prepocessing: ../input/brown_train/cj53
Prepocessing: ../input/brown_train/cl05
Prepocessing: ../input/brown_train/cf02
Prepocessing: ../input/brown_train/cn01
Prepocessing: ../input/brown_train/cj65
Prepocessing: ../input/brown_train/cf05
Prepocessing: ../input/brown_train/cj62
Prepocessing: ../input/brown_train/cn06
Prepocessing: ../input/brown_train/cd08
Prepocessing: ../input/brown_train/cj01
Prepocessing: ../input/brown_train/cb02
Prepocessing: ../input/brown_train/cj06
Prepocessing: ../input/brown_train/cb05
Prepocessing: ../input/brown_train/cj39
Prepocessing: ../input/brown_train/cj30
Prepocessing: ../input/brown_train/ch02
```

```
Prepocessing: ../input/brown_train/ch05
Prepocessing: ../input/brown_train/cj37
Prepocessing: ../input/brown_train/cj08
Prepocessing: ../input/brown_train/cf04
Prepocessing: ../input/brown_train/cj63
Prepocessing: ../input/brown_train/cd09
Prepocessing: ../input/brown_train/cn07
Prepocessing: ../input/brown_train/cf03
Prepocessing: ../input/brown_train/cj64
Prepocessing: ../input/brown_train/cd07
Prepocessing: ../input/brown_train/cn09
Prepocessing: ../input/brown_train/cf35
Prepocessing: ../input/brown_train/cj52
Prepocessing: ../input/brown_train/cl04
Prepocessing: ../input/brown_train/cf32
Prepocessing: ../input/brown_train/cl03
Prepocessing: ../input/brown_train/cj55
Prepocessing: ../input/brown_train/ck27
Prepocessing: ../input/brown_train/ca29
Prepocessing: ../input/brown_train/cg40
Prepocessing: ../input/brown_train/ca16
Prepocessing: ../input/brown_train/ck18
Prepocessing: ../input/brown_train/ck20
Prepocessing: ../input/brown_train/ca11
Prepocessing: ../input/brown_train/cg47
Prepocessing: ../input/brown_train/ca18
Prepocessing: ../input/brown_train/ck16
Prepocessing: ../input/brown_train/cg71
Prepocessing: ../input/brown_train/cc15
Prepocessing: ../input/brown_train/ck29
Prepocessing: ../input/brown_train/ca27
Prepocessing: ../input/brown_train/cg49
Prepocessing: ../input/brown_train/ck11
Prepocessing: ../input/brown_train/ca20
Prepocessing: ../input/brown_train/cc12
Prepocessing: ../input/brown_train/ce10
Prepocessing: ../input/brown_train/cg22
Prepocessing: ../input/brown_train/cr07
Prepocessing: ../input/brown_train/cg25
Prepocessing: ../input/brown_train/ce17
Prepocessing: ../input/brown_train/ce28
Prepocessing: ../input/brown_train/cp04
Prepocessing: ../input/brown_train/cg13
Prepocessing: ../input/brown_train/ce21
Prepocessing: ../input/brown_train/cr09
Prepocessing: ../input/brown_train/ce26
Prepocessing: ../input/brown_train/cg14
Prepocessing: ../input/brown_train/ca42
```

```
Prepocessing: ../input/brown_train/cp03
Prepocessing: ../input/brown_train/cg48
Prepocessing: ../input/brown_train/ck10
Prepocessing: ../input/brown_train/cc13
Prepocessing: ../input/brown_train/ca21
Prepocessing: ../input/brown_train/ck17
Prepocessing: ../input/brown_train/ca19
Prepocessing: ../input/brown_train/cg70
Prepocessing: ../input/brown_train/ca26
Prepocessing: ../input/brown_train/ck28
Prepocessing: ../input/brown_train/cc14
Prepocessing: ../input/brown_train/ck21
Prepocessing: ../input/brown_train/ca10
Prepocessing: ../input/brown_train/cg46
Prepocessing: ../input/brown_train/ca28
Prepocessing: ../input/brown_train/ck26
Prepocessing: ../input/brown_train/cg41
Prepocessing: ../input/brown_train/ck19
Prepocessing: ../input/brown_train/ca17
Prepocessing: ../input/brown_train/cp02
Prepocessing: ../input/brown_train/cg15
Prepocessing: ../input/brown_train/ce27
Prepocessing: ../input/brown_train/ca43
Prepocessing: ../input/brown_train/ce18
Prepocessing: ../input/brown_train/ca44
Prepocessing: ../input/brown_train/ce20
Prepocessing: ../input/brown_train/cg12
Prepocessing: ../input/brown_train/cp05
Prepocessing: ../input/brown_train/cr08
Prepocessing: ../input/brown_train/ce16
Prepocessing: ../input/brown_train/cg24
Prepocessing: ../input/brown_train/cr01
Prepocessing: ../input/brown_train/ce29
Prepocessing: ../input/brown_train/cr06
Prepocessing: ../input/brown_train/cg23
Prepocessing: ../input/brown_train/ce11
Prepocessing: ../input/brown_train/cp29
Prepocessing: ../input/brown_train/cp16
Prepocessing: ../input/brown_train/ce33
Prepocessing: ../input/brown_train/cg01
Prepocessing: ../input/brown_train/cg39
Prepocessing: ../input/brown_train/cg06
Prepocessing: ../input/brown_train/ce34
Prepocessing: ../input/brown_train/cp11
Prepocessing: ../input/brown_train/cm01
Prepocessing: ../input/brown_train/cp18
Prepocessing: ../input/brown_train/cg30
Prepocessing: ../input/brown_train/ce02
```

```
Prepocessing: ../input/brown_train/cp27
Prepocessing: ../input/brown_train/cg08
Prepocessing: ../input/brown_train/cm06
Prepocessing: ../input/brown_train/cp20
Prepocessing: ../input/brown_train/ce05
Prepocessing: ../input/brown_train/cg37
Prepocessing: ../input/brown_train/ca35
Prepocessing: ../input/brown_train/cg63
Prepocessing: ../input/brown_train/ck04
Prepocessing: ../input/brown_train/cg64
Prepocessing: ../input/brown_train/ca32
Prepocessing: ../input/brown_train/ck03
Prepocessing: ../input/brown_train/ca04
Prepocessing: ../input/brown_train/cg52
Prepocessing: ../input/brown_train/cc09
Prepocessing: ../input/brown_train/cg55
Prepocessing: ../input/brown_train/ca03
Prepocessing: ../input/brown_train/cg09
Prepocessing: ../input/brown_train/cg36
Prepocessing: ../input/brown_train/ce04
Prepocessing: ../input/brown_train/cp21
Prepocessing: ../input/brown_train/cp19
Prepocessing: ../input/brown_train/cp26
Prepocessing: ../input/brown_train/ce03
Prepocessing: ../input/brown_train/cg31
Prepocessing: ../input/brown_train/cg38
Prepocessing: ../input/brown_train/cp10
Prepocessing: ../input/brown_train/ce35
Prepocessing: ../input/brown_train/cg07
Prepocessing: ../input/brown_train/cp28
Prepocessing: ../input/brown_train/ce32
Prepocessing: ../input/brown_train/cp17
Prepocessing: ../input/brown_train/cg54
Prepocessing: ../input/brown_train/ca02
Prepocessing: ../input/brown_train/ca05
Prepocessing: ../input/brown_train/cg53
Prepocessing: ../input/brown_train/cc08
Prepocessing: ../input/brown_train/cg65
Prepocessing: ../input/brown_train/ca33
Prepocessing: ../input/brown_train/cc01
Prepocessing: ../input/brown_train/ck02
Prepocessing: ../input/brown_train/cc06
Prepocessing: ../input/brown_train/cg62
Prepocessing: ../input/brown_train/ck05
Prepocessing: ../input/brown_train/cj77
Prepocessing: ../input/brown_train/cn13
Prepocessing: ../input/brown_train/cf10
Prepocessing: ../input/brown_train/cj48
```

```
Prepocessing: ../input/brown_train/cn14
Prepocessing: ../input/brown_train/cf28
Prepocessing: ../input/brown_train/cj70
Prepocessing: ../input/brown_train/cf17
Prepocessing: ../input/brown_train/cl19
Prepocessing: ../input/brown_train/cj46
Prepocessing: ../input/brown_train/cl10
Prepocessing: ../input/brown_train/cn22
Prepocessing: ../input/brown_train/cf21
Prepocessing: ../input/brown_train/cd13
Prepocessing: ../input/brown_train/cj79
Prepocessing: ../input/brown_train/cn25
Prepocessing: ../input/brown_train/cf19
Prepocessing: ../input/brown_train/cl17
Prepocessing: ../input/brown_train/cj41
Prepocessing: ../input/brown_train/cf26
Prepocessing: ../input/brown_train/ch18
Prepocessing: ../input/brown_train/cb16
Prepocessing: ../input/brown_train/cj15
Prepocessing: ../input/brown_train/ch27
Prepocessing: ../input/brown_train/cb11
Prepocessing: ../input/brown_train/ch20
Prepocessing: ../input/brown_train/cj12
Prepocessing: ../input/brown_train/cb27
Prepocessing: ../input/brown_train/ch29
Prepocessing: ../input/brown_train/cf43
Prepocessing: ../input/brown_train/ch16
Prepocessing: ../input/brown_train/cb18
Prepocessing: ../input/brown_train/cj24
Prepocessing: ../input/brown_train/cf44
Prepocessing: ../input/brown_train/cb20
Prepocessing: ../input/brown_train/cj23
Prepocessing: ../input/brown_train/ch11
Prepocessing: ../input/brown_train/cl16
Prepocessing: ../input/brown_train/cf18
Prepocessing: ../input/brown_train/cn24
Prepocessing: ../input/brown_train/cj40
Prepocessing: ../input/brown_train/cf27
Prepocessing: ../input/brown_train/cd15
Prepocessing: ../input/brown_train/cj47
Prepocessing: ../input/brown_train/cn23
Prepocessing: ../input/brown_train/cl11
Prepocessing: ../input/brown_train/cd12
Prepocessing: ../input/brown_train/cf20
Prepocessing: ../input/brown_train/cj78
Prepocessing: ../input/brown_train/cf29
Prepocessing: ../input/brown_train/cn15
Prepocessing: ../input/brown_train/cj71
```

```
Prepocessing: ../input/brown_train/cl18
Prepocessing: ../input/brown_train/cf16
Prepocessing: ../input/brown_train/cj76
Prepocessing: ../input/brown_train/cl20
Prepocessing: ../input/brown_train/cn12
Prepocessing: ../input/brown_train/cf11
Prepocessing: ../input/brown_train/cj49
Prepocessing: ../input/brown_train/cf45
Prepocessing: ../input/brown_train/cb21
Prepocessing: ../input/brown_train/ch10
Prepocessing: ../input/brown_train/cj22
Prepocessing: ../input/brown_train/ch28
Prepocessing: ../input/brown_train/cb26
Prepocessing: ../input/brown_train/cf42
Prepocessing: ../input/brown_train/cj25
Prepocessing: ../input/brown_train/cb19
Prepocessing: ../input/brown_train/ch17
Prepocessing: ../input/brown_train/cb10
Prepocessing: ../input/brown_train/cj13
Prepocessing: ../input/brown_train/ch21
Prepocessing: ../input/brown_train/cb17
Prepocessing: ../input/brown_train/ch19
Prepocessing: ../input/brown_train/ch26
Prepocessing: ../input/brown_train/cj14
 Number of Sentences: 55684, Word list count: 1126281, Tag list count: 1126281, Transition Tag
### Replace word tokens <= 5 with 'UNK
Number of tokens after replacement: Word - 1126281, Word_Tag - 1126281
```

4.1 Obtain frequency counts from the collection of all the training files (counted together). You will need the following types of frequency counts: word-tag counts, tag un-igram counts, and tag bigram counts. Let's denote these by C(wi, ti), C(ti) and C(ti1, ti) respectively. Report these quantities in different output files.

```
In [5]: def get_unigrams(token_list):
            tokens_with_count = Counter(token_list)
            return tokens_with_count

In [6]: def get_ngrams(token_list, n):
            ngrams_zip = zip(*[token_list[i:] for i in range(n)])
            ngrams_list = [" ".join(element) for element in ngrams_zip]
            ngrams_keys_counts = Counter(ngrams_list)
            return ngrams_keys_counts

In [7]: print("Unigragms for words")
        train_word_unigrams = get_unigrams(train_word_tokens)
        print(len(train_word_unigrams))
        vocab_size = len(train_word_unigrams)
        print("Vocab size {}".format(vocab_size))
```

```
        print("Unigragms for word-tag")
        print(len(train_word_tag_tokens))
        train_word_tag_unigrams = get_unigrams(train_word_tag_tokens)
        with open('../output/word_tag_unigrams.txt', 'w') as word_tag_unigrams_output_file:
            word_tag_unigrams_output_file.write(str(train_word_tag_unigrams))
        print(len(train_word_tag_unigrams))

        print("Unigragms for tags")
        train_tag_unigrams = get_unigrams(train_tag_tokens)
        print(len(train_tag_unigrams))

        print("Unigrams for transition tags")
        train_transition_tag_unigrams = get_unigrams(train_transition_tag_tokens)
        with open('../output/tag_unigrams.txt', 'w') as transition_tag_unigrams_output_file:
            transition_tag_unigrams_output_file.write(str(train_transition_tag_unigrams))
        print(len(train_transition_tag_unigrams))

        print("Bigrams for transition tags")
        train_transition_tag_bigrams = get_ngrams(train_transition_tag_tokens, 2)
        with open('../output/tag_bigrams.txt', 'w') as transition_tag_bigrams_output_file:
            transition_tag_bigrams_output_file.write(str(train_transition_tag_bigrams))
        print(len(train_transition_tag_bigrams))
```

```
Unigragms for words
30036
Vocab size 30036
Unigragms for word-tag
1126281
41029
Unigragms for tags
470
Unigrams for transition tags
472
Bigrams for transition tags
8255
```

4.2

A transition probability is the probability of a tag given its previous tag. Calculate transition probabilities of the training set using the following equation:
P(ti1, ti) = C(ti1, ti)/C(ti1)

```
In [8]: def get_transition_probability(tag_unigrams, tag_bigrams, lambda_value, vocab_size):
            transition_probability = {}
            for i in tag_bigrams:
                previous_tag = i.split(" ")[0]
                transition_probability[i] = (tag_bigrams[i] + lambda_value) / (tag_unigrams[pr
            return transition_probability
```

```
In [9]: print("Get transition probability")
        train_transition_probability = get_transition_probability(train_transition_tag_unigrams
        # print(train_transition_probability)
```

```
Get transition probability
```

4.3

An emission probability is the probability of a given word being associated with a given tag. Calculate emission probabilities of the training set using the following equation:
P(wi, ti) = C(wi, ti)/C(ti)

```
In [10]: def get_emission_probability(word_tag_unigrams, tag_unigrams, lambda_value, vocab_size
             emission_probability = {}
             for i in word_tag_unigrams:
                 split_tokens = i.split('/')
                 word = split_tokens[0]
                 tag = split_tokens[-1]
                 emission_probability[i] = (word_tag_unigrams[i] + lambda_value) / (tag_unigram
         #         print(word, tag)
         #         print(word_tag_unigrams[i], lambda_value)
         #         print(tag_unigrams[tag], lambda_value, vocab_size)
         #         print(emission_probability[i])
             return emission_probability
```

```
In [11]: print("Get emission probability")
         train_emission_probability = get_emission_probability(train_word_tag_unigrams, train_t
         # print(train_emission_probability)
```

```
Get emission probability
```

4.4 Generate 5 random sentences using the previously learned HMM. Output each sentence (with the POS tags) and its probability of being generated.

Hint: With the help of emission probabilities and transition probabilities collected from 4.2 and 4.3, 1. Start with ‘’ tag. 2. Choose next tag based on random choice but considering probabilities e.g. tag draw = random.choices(„). 3. Now choose word for the corresponding tag using emission probabilities (all the words that can be generated from that tag and corresponding probabilities they can be generated with.)e.g. word draw = random.choices(„). 4. Keep repeating steps 2 and 3 till you hit end token ‘.’ 5. Report the sentence and the probability with which this sentence can be generated.

```
In [12]: def generate_sentence(transition_probability, emission_probability):

             sentence_tags = []
             sentence_words = []
             sentence_transition_probability = []
             sentence_emission_probability = []
```

```
    while(True):
        # Get potetial next set of tags and its probabilities
        current_tag = "START"
        next_tag = []
        next_tag_probabilities = []
        next_tag_dict = {}
        for i in transition_probability:
            if i.split(" ")[0] == current_tag and i != "END START":
                tag = i.split(" ")[1]
                probability = transition_probability[i]
                next_tag.append(tag)
                next_tag_probabilities.append(probability)
                next_tag_dict[tag] = probability

        # Random pick of a tag
        tag_drawn = random.choices(next_tag, next_tag_probabilities)[0]

#         print("Tag")
#         print(next_tag)
#         print(next_tag_probabilities)
#         print(tag_drawn)

        # Limiting number of words in the sentence to 30
        if (tag_drawn == "END") | (len(sentence_words)==30):
            break

        # Get potential words to be chosen out of a selected tag
        next_word = []
        next_word_probabilities = []
        next_word_dict = {}
        for i in emission_probability:
            if i.split("/")[1] == tag_drawn:
                word = i.split("/")[0]
                probability = emission_probability[i]
                next_word.append(word)
                next_word_probabilities.append(probability)
                next_word_dict[word] = probability

        # Random pick of a word
#         print("Word")
#         print(next_word)
#         print(next_word_probabilities)
        word_drawn = random.choices(next_word, next_word_probabilities)[0]

        sentence_tags.append(tag_drawn)
        sentence_words.append(word_drawn)
        sentence_transition_probability.append(next_tag_dict[tag_drawn])
        sentence_emission_probability.append(next_word_dict[word_drawn])
```

```
            current_tag = next_tag

        total_probability = 1
        for i in range(len(sentence_words)):
            total_probability = total_probability * sentence_transition_probability[i] * s

        sentence = sentence_words[0]
        for i in range(1, len(sentence_words)):
            sentence = sentence + " " + sentence_words[i]
        return(sentence_tags, sentence_words, sentence_transition_probability, sentence_em
```

```
In [13]: sentence_generation_file = open('../output/generated_sentences.txt', 'w')
         for i in range(1,6):
             print("Generating sentence: {}".format(i))
             sentence_generation_file.write("Sentence: {}\n".format(1))
             sentence_tags, sentence_words, sentence_transition_probability, \
                 sentence_emission_probability, total_probability, sentence = generate_sentence
                     train_transition_probability, train_emission_probability
                 )
             sentence_generation_file.write("Words: \n {} \n ".format(sentence_words))
             sentence_generation_file.write("Tags: \n {} \n ".format(sentence_tags))
             sentence_generation_file.write("Transition probability \n {} \n ".format(sentence_
             sentence_generation_file.write("Emission probability \n {} \n ".format(sentence_em
             sentence_generation_file.write("Total probability: {} \n \n".format(total_probabil
             sentence_generation_file.write("Sentence: \n {} \n \n".format(sentence))
         sentence_generation_file.close()
```

```
Generating sentence: 1
Generating sentence: 2
Generating sentence: 3
Generating sentence: 4
Generating sentence: 5
```

4.5

For each word in the test dataset, derive the most probable POS tag sequence using the Viterbi algorithm; pseudo-code can be found in the textbook http://web.stanford.edu/ jurafsky/slp3/ed3book.pdf under Figure 8.5. Viterbi algorithm should be implemented following the pseudocode provided for reference.

Hint: Traversing through back-pointer data structure at the end of algorithm would provide information about the best possible previous tag. So when you are at the second last word of the sentence, calling back-pointer here would give the tag information for the first word in the sentence.

Submit the output in a file exactly with the following format (where each line contains no more than one pair): < sentenceID = 1 > word/tag word/tag .... word/tag < EOS > < sentenceID = 2 > word/tag word/tag .... word/tag < EOS >

```
In [14]: def get_state_transition_matrix(states, transition_probability):
             state_transition_df = pd.DataFrame(0, index=states, columns=states)
```

```
            for key in transition_probability:
                state1, state2 = key.split(" ")
                state_transition_df.loc[state1, state2] = transition_probability[key]
            return state_transition_df

In [15]: def get_word_state_emission_matrix(words, states, emission_probability):
            word_state_emission_df = pd.DataFrame(0, index=words, columns=states)
            for key in emission_probability:
                split_tokens = key.split('/')
                word = split_tokens[0]
                tag = split_tokens[-1]
                word_state_emission_df.loc[word, tag] = emission_probability[key]
            return word_state_emission_df

In [16]: def viterbi_algorithm(words, states, state_transition_df, word_state_emission_df):
            score_df = pd.DataFrame(0, index=range(len(words)), columns=states)
            trace_df = pd.DataFrame("", index=range(len(words)), columns=states)
            for i in range(len(words)):
                # Check for word existence
                word = words[i]
        #         print("########## Word: {}".format(words[i]))
                if word not in word_state_emission_df.index:
                    print("Converting word: {} to UNK".format(word))
                    word = 'UNK'

                # For initial word
                if i == 0:
                    temp_word_state_emission_list = word_state_emission_df.loc[word, states].
                    temp_state_transition_df_list = state_transition_df.loc["START", states].
                    temp_score = [temp_word_state_emission_list[i]*temp_state_transition_df_li
                    score_df.loc[i, states] = temp_score
                    trace_df.loc[i, states] = ["START"] * len(states)
        #             print(score_df.loc[i, :][score_df.loc[i, :] > 0])
        #             print(">>>>>>>>>>>>")
        #             print(trace_df.loc[i, :][score_df.loc[i, :] > 0])
        #             print(">>>>>>>>>>>>")
                else:
                    previous_row_probability_list = score_df.loc[i-1, :].values.tolist()
                    for j in range(len(states)):
                        current_state = states[j]
                        temp_word_state_emission_list = [word_state_emission_df.loc[word, curr
                        temp_state_transition_df_list = state_transition_df.loc[states, curren
                        temp_score_list = [
                            previous_row_probability_list[m] * temp_word_state_emission_list[m
                            for m in range(len(temp_word_state_emission_list))
                        ]
                        f = lambda k: temp_score_list[k]
                        arg_max_temp_score = max(range(len(temp_score_list)), key=f)
```

```
                    score_df.loc[i, current_state] = temp_score_list[arg_max_temp_score]
                    trace_df.loc[i, current_state] = states[arg_max_temp_score]
            #             print(score_df.loc[i, :][score_df.loc[i, :] > 0])
            #             print(">>>>>>>>>>>>")
            #             print(trace_df.loc[i, :][score_df.loc[i, :] > 0])
            #             print(">>>>>>>>>>>>")

                # Trace back
                result = []
                # Get the highest score for the last word
                chosen_state = score_df.loc[len(words)-1, :].argmax()
                result.append((words[len(words)-1], chosen_state))
                trace_back_column = trace_df.loc[len(words)-1, chosen_state]
                for i in range(len(words)-2, -1, -1):
                    result.append((words[i], trace_back_column))
                    trace_back_column = trace_df.loc[i, trace_back_column]
                return result[::-1]
```

In [17]:
```
def get_test_data(file_name):
    file_pointer = open(file_name, "r")
    sentences = []
    sentence = []
    for line in file_pointer:

        # Remove duplicate spaces
        file_line_content = re.sub(' +', ' ', line)

        # Strip of begin and end spaces
        file_line_content = file_line_content.strip()

        # If line is not empty
        if file_line_content != "":
            if "sentence ID" in file_line_content:
                continue
            elif  "EOS" in file_line_content:
                sentences.append(sentence)
                sentence = []
            else:
                sentence.append(file_line_content)

    file_pointer.close()
    return sentences
```

In [18]:
```
state_transition_df = get_state_transition_matrix(list(train_transition_tag_unigrams.
del state_transition_df['START']
```

In [20]:
```
word_state_emission_df = get_word_state_emission_matrix(list(train_word_unigrams.keys
```

In [21]:
```
test_sentences = get_test_data('../input/Test_File.txt')
viterbi_output_file = open('../output/viterbi_output.txt', 'w')
```

```
        for i in range(len(test_sentences)):
            print("Processing sentence {}".format(i))
            viterbi_output_file.write("< sentence ID = {} >\n".format(i+1))
        #     test_sentences[i] = ['Bella', 'wanted', 'to', 'board', 'the', 'bus', 'to', 'Chi
        #     test_sentences[i] = ['John', 'nailed', 'the', 'board', 'over', 'the', 'window']
            result = viterbi_algorithm(test_sentences[i],
                                       list(train_tag_unigrams.keys()),
                                       state_transition_df,
                                       word_state_emission_df
                                   )
            for word,tag in result:
                viterbi_output_file.write("{}/{}\n".format(word,tag))
            viterbi_output_file.write("< EOS >\n")
        viterbi_output_file.close()
```

```
Processing sentence 0
Converting word: kilowatt-hour to UNK
Converting word: kilowatts to UNK
Converting word: kilowatt to UNK
Converting word: $8 to UNK


/Users/karangm/PycharmProjects/pos_tagging/venv/lib/python3.6/site-packages/ipykernel_launcher
The current behaviour of 'Series.argmax' is deprecated, use 'idxmax'
instead.
The behavior of 'argmax' will be corrected to return the positional
maximum in the future. For now, use 'series.values.argmax' or
'np.argmax(np.array(values))' to get the position of the maximum
row.


Processing sentence 1
Converting word: kilowatt-hour to UNK
Processing sentence 2
Processing sentence 3
Converting word: out-of-pocket to UNK
Processing sentence 4
Converting word: electric-utility to UNK
Processing sentence 5
Processing sentence 6
Processing sentence 7
Converting word: utility-cost to UNK
Processing sentence 8
Processing sentence 9
Converting word: subtype to UNK
Converting word: distributes to UNK
Processing sentence 10
Processing sentence 11
```

```
Converting word: whereof to UNK
Converting word: hereunto to UNK
Converting word: 11th to UNK
Converting word: sixty-one to UNK
Converting word: eighty-sixth to UNK
Processing sentence 12
Converting word: Resolution to UNK
Converting word: 22nd to UNK
Converting word: Maritime to UNK
Processing sentence 13
Converting word: intermissions to UNK
Processing sentence 14
Processing sentence 15
Converting word: whereof to UNK
Converting word: hereunto to UNK
Converting word: sixty-one to UNK
Converting word: eighty-sixth to UNK
Processing sentence 16
Processing sentence 17
Converting word: frugality to UNK
Processing sentence 18
Processing sentence 19
Converting word: Pilgrims to UNK
Processing sentence 20
Processing sentence 21
Converting word: Crombie to UNK
Converting word: Blatz's to UNK
Processing sentence 22
Converting word: Crombie to UNK
Processing sentence 23
Converting word: Blatz to UNK
Converting word: Smithtown to UNK
Processing sentence 24
Processing sentence 25
Processing sentence 26
Converting word: pegboard to UNK
Processing sentence 27
Processing sentence 28
Processing sentence 29
Processing sentence 30
Processing sentence 31
Processing sentence 32
Converting word: Mattie to UNK
Converting word: Toonker to UNK
Converting word: Burkette to UNK
Converting word: yanking to UNK
Processing sentence 33
Processing sentence 34
```

```
Processing sentence 35
Processing sentence 36
Processing sentence 37
Converting word: tramp to UNK
Processing sentence 38
Converting word: spellbound to UNK
Processing sentence 39
Processing sentence 40
Processing sentence 41
Processing sentence 42
Converting word: Juanita to UNK
Converting word: Lattimer to UNK
Processing sentence 43
Converting word: Randolph to UNK
Converting word: Joel to UNK
Converting word: replanted to UNK
Converting word: Annie to UNK
Processing sentence 44
Processing sentence 45
Processing sentence 46
Processing sentence 47
Processing sentence 48
Converting word: Juanita's to UNK
Processing sentence 49
Processing sentence 50
Converting word: 4,585 to UNK
Converting word: Fisk to UNK
Processing sentence 51
Processing sentence 52
Processing sentence 53
Converting word: benchmarks to UNK
Converting word: profoundity to UNK
Processing sentence 54
Processing sentence 55
Converting word: libertarian to UNK
Processing sentence 56
Converting word: inalienable to UNK
Processing sentence 57
Converting word: Avowed to UNK
Converting word: freethinkers to UNK
Processing sentence 58
Converting word: traditionalistic to UNK
Processing sentence 59
Converting word: socially-oriented to UNK
Processing sentence 60
Processing sentence 61
Processing sentence 62
Processing sentence 63
```

```
Processing sentence 64
Converting word: codified to UNK
Processing sentence 65
Processing sentence 66
Converting word: Hesperus to UNK
Converting word: Lucifer to UNK
Processing sentence 67
Processing sentence 68
Converting word: Hesperus to UNK
Processing sentence 69
Processing sentence 70
Processing sentence 71
Processing sentence 72
Processing sentence 73
Converting word: Warmly to UNK
Processing sentence 74
Processing sentence 75
Converting word: SX-21 to UNK
Processing sentence 76
Converting word: plain-clothesmen to UNK
Processing sentence 77
Converting word: Thor's to UNK
Converting word: Antony to UNK
Converting word: zing to UNK
Processing sentence 78
Processing sentence 79
Processing sentence 80
Converting word: Ought to UNK
Converting word: edifying to UNK
Converting word: Trial to UNK
Converting word: anti-Semites to UNK
Converting word: skull-bashings to UNK
Converting word: gassings to UNK
Processing sentence 81
Processing sentence 82
Converting word: patriots to UNK
Converting word: terrorizing to UNK
Converting word: meanest to UNK
Converting word: pulverizing to UNK
Processing sentence 83
Converting word: Trial to UNK
Converting word: anti-Semitic to UNK
Converting word: demoralization to UNK
Processing sentence 84
Processing sentence 85
Converting word: Wansee to UNK
Converting word: Heydrich to UNK
Processing sentence 86
```

```
Converting word: Trial to UNK
Processing sentence 87
Converting word: Trial to UNK
Converting word: anti-Semitism to UNK
Processing sentence 88
Converting word: anti-Semitism to UNK
Converting word: Jew-baiter to UNK
Processing sentence 89
Converting word: Heydrich to UNK
Converting word: Goering to UNK
Converting word: Solution to UNK
Converting word: strangulation to UNK
Converting word: emigration to UNK
Processing sentence 90
Converting word: casualties to UNK
Processing sentence 91
Converting word: DePugh to UNK
Converting word: Lauchli to UNK
Processing sentence 92
Converting word: Minutemen to UNK
Processing sentence 93
Processing sentence 94
Converting word: Vietnam to UNK
Processing sentence 95
Converting word: Albanians to UNK
Processing sentence 96
Converting word: Malinovsky to UNK
Converting word: exalting to UNK
Processing sentence 97
Processing sentence 98
Processing sentence 99
Converting word: liberating to UNK
Processing sentence 100
Converting word: squashed to UNK
Converting word: suntan to UNK
Converting word: semi-inflated to UNK
Processing sentence 101
Processing sentence 102
Processing sentence 103
Processing sentence 104
Converting word: dirt-catcher to UNK
Processing sentence 105
Processing sentence 106
Converting word: out-of-sight to UNK
Converting word: out-of-mind to UNK
Converting word: trek to UNK
Processing sentence 107
Converting word: Soignee to UNK
```

```
Processing sentence 108
Processing sentence 109
Processing sentence 110
Processing sentence 111
Converting word: Jannequin's to UNK
Converting word: tarantara to UNK
Converting word: rum-tum-tum to UNK
Converting word: boom-boom-boom to UNK
Converting word: chansons to UNK
Converting word: Jannequin to UNK
Converting word: Lassus to UNK
Processing sentence 112
Converting word: Jean-Marie to UNK
Converting word: LeClair to UNK
Converting word: Bodin to UNK
Converting word: Beismortier to UNK
Converting word: Corrette to UNK
Converting word: Mondonville to UNK
Processing sentence 113
Converting word: forego to UNK
Processing sentence 114
Converting word: out-of-the-way to UNK
Processing sentence 115
Converting word: dancelike to UNK
Processing sentence 116
Converting word: Elegance to UNK
Processing sentence 117
Processing sentence 118
Converting word: Alvise to UNK
Processing sentence 119
Processing sentence 120
Processing sentence 121
Processing sentence 122
Converting word: Disapproval to UNK
Processing sentence 123
Converting word: full-dress to UNK
Processing sentence 124
Processing sentence 125
Processing sentence 126
Processing sentence 127
Processing sentence 128
Processing sentence 129
Processing sentence 130
Converting word: Stacy to UNK
Converting word: Forbes to UNK
Processing sentence 131
Processing sentence 132
Processing sentence 133
```

```
Processing sentence 134
Converting word: Kimball to UNK
Converting word: Stacy to UNK
Processing sentence 135
Processing sentence 136
Processing sentence 137
Converting word: Soak to UNK
Processing sentence 138
Converting word: gullet to UNK
Processing sentence 139
Converting word: Stacy to UNK
Converting word: remarry to UNK
Converting word: Forbes to UNK
Processing sentence 140
Processing sentence 141
Converting word: Methodism to UNK
Processing sentence 142
Converting word: Incurably to UNK
Converting word: devout to UNK
Converting word: Greenleaf to UNK
Converting word: Whittier to UNK
Converting word: 1807-1892 to UNK
Converting word: plenary to UNK
Processing sentence 143
Converting word: Oberlin to UNK
Processing sentence 144
Processing sentence 145
Processing sentence 146
Converting word: 1811-1884 to UNK
Converting word: Lyman to UNK
Converting word: Beecher to UNK
Processing sentence 147
Processing sentence 148
Converting word: anti-slavery to UNK
Converting word: Finney to UNK
Converting word: revivals to UNK
Processing sentence 149
```