<ASH>NLP_Project_Round1_Report

October 20, 2019

0.1 Natural Language Processing of a Book

Team Name: ASH
Team Members: Ayush Gupta(17ucs042), Harshit Garg(17ucs064), Shubhi Rustagi(17ucs158)

0.2 Book Details

Book Title - Off Sandy Hook, and other stories Author's Name - Richard Dehan Subjects - Fiction, Short stories Number of Sentences - 11202 Number of Words - 98822 Character set encoding - UTF-8

This is part of NLP-Projects Phase-1. In this project, we have used the following libraries and toolkits: * **nltk** - A leading platform in Python designed to provide easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries. These libraries are used for classification, tokenisation, lemmatisation etc. * **string** - This module consists of basic operations that can be performed on string. * **WordNetLemmatizer** - It is a package for lemmatization. Lemmatization is the process of grouping together the words having similar meaning to a particular word. * **stopwords** - Set of commonly used words that are mostly ignored in text processing. * **matplotlib** - Library used for data visualisation. In other words, it is used for plotting the relationship among different components. * **re** - Library used for operations on regular expressions. * **wordcloud** - Library is used to plot the relative frequency of the words in text.

To begin, we import all the necssary libraries.

[18]:
```
import nltk
import string
from nltk.stem import WordNetLemmatizer
from nltk.corpus import stopwords
from nltk.corpus import brown
from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt
import re
```

0.3 Preprocessing

Now, open the book and read it in a variable *file*. Further, we read the contents of the book in another variable *T*. We declare a variable *test* equal to *T* in case we require the original text later in the process as *T* will undergo text processing.

[19]:
```
#Reading Book
file = open("Sandy_Hook_NLP.txt",encoding="utf8")

# variable T
T = file.read()

# variable test
test=T

# Printing the contents of the book stored in T
print(T[:1000])
```

```
Project Gutenberg's Off Sandy Hook and other stories, by Richard Dehan

This eBook is for the use of anyone anywhere in the United States and most
other parts of the world at no cost and with almost no restrictions
whatsoever.  You may copy it, give it away or re-use it under the terms of
the Project Gutenberg License included with this eBook or online at
www.gutenberg.org.  If you are not located in the United States, you'll have
to check the laws of the country where you are located before using this ebook.

Title: Off Sandy Hook and other stories

Author: Richard Dehan

Release Date: October 8, 2019 [EBook #60452]

Language: English

Character set encoding: UTF-8

*** START OF THIS PROJECT GUTENBERG EBOOK OFF SANDY HOOK AND OTHER STORIES ***



Produced by Richard Tonsing and the Online Distributed
Proofreading Team at http://www.pgdp.net (This file was
produced from images generously made available by The
Internet Archive)
```

OFF SANDY HOOK

Further, punctuation marks are removed using maketrans() function and translate() function. First, *maketrans()* maps the string.punctuation to None and creates a mapping table. Then, using *translate()*, all the punctuations in *T* are replaced by None according to the mapping table. This is done in order to obtain the meaningful words.

[20]:
```
translator = str.maketrans("","",string.punctuation)
T=T.translate(translator)
print(T[:1000])
```

```
Project Gutenbergs Off Sandy Hook and other stories by Richard Dehan

This eBook is for the use of anyone anywhere in the United States and most
other parts of the world at no cost and with almost no restrictions
whatsoever  You may copy it give it away or reuse it under the terms of
the Project Gutenberg License included with this eBook or online at
wwwgutenbergorg  If you are not located in the United States youll have
to check the laws of the country where you are located before using this ebook

Title Off Sandy Hook and other stories

Author Richard Dehan

Release Date October 8 2019 EBook 60452

Language English

Character set encoding UTF8

 START OF THIS PROJECT GUTENBERG EBOOK OFF SANDY HOOK AND OTHER STORIES



Produced by Richard Tonsing and the Online Distributed
Proofreading Team at httpwwwpgdpnet This file was
produced from images generously made available by The
Internet Archive
```

OFF SANDY HOOK

AND OTHER

Removal of the acknowledgement section and transcriber's note is done by using the substitution function and regular expressions.

[21]:
```
# Removing acknowledgement
T = re.sub("Project[\s\S]*CONTENTS","",T)

# Removing the last part(Transciber's Notes)
T = re.sub("TRANSCRIBER[\s\S]*","",T)
print(T[:1000])
```

```
                                PAGE
OFF SANDY HOOK                     1

GEMINI                            15

A DISH OF MACARONI                31

FREDDY  CIE                44

UNDER THE ELECTRICS               60

VALCOURTS GRIN                68

THE EVOLUTION OF THE FAIREST   81

THE REVOLT OF RUSTLETON           95

A DYSPEPTICS TRAGEDY          107

RENOVATION                        119

THE BREAKING PLACE                133

A LANCASHIRE DAISY                143

A PITCHED BATTLE                  154

THE TUG OF WAR                    164
```

```
GAS                              180

AIR                               193

SIDE                             205

A SPIRIT ELO
```

Similarly as above, we remove the chapter names from the index of the book.

[22]:
```
# Removing Chapter Names
T=re.sub("[A-Z]{2,}","",T)
print(T[:1000])
```

```
                              1

                                   15

A                  31

                     44

                   60

S                    68

          81

               95

A S              107

                          119

                   133

A                 143

A                   154

                      164

                                180

                                 193
```

205

A 219

S 230

241

S 264

S 276

287

Now, we remove the Chapter Numbers using the substitution function and regular expression.

[23]:
```
# Removing Chapter Numbers
T=re.sub("[0-9]+","",T)
print(T[:1000])
```

A

S

A S

A

A

A

S

S

S

A

Conversion of the text to lower case for the ease of analysis.

[24]:
```
# Converting text to lowercase
T = T.lower()
print(T[:1000])
```

a

s

a s

a

a

a

s

s

s

a

[25]:
```
# splitting the lines and joining them into one.
T = " ".join(T.split())
wordcloudT=T
print(T[:1000])
```

```
 a   s  a s a a a s s s a   on board the rampatina liner eleven
days and a half out from liverpool the usual terrific sensation created by the
```

```
appearance of the pilotyacht prevailed necks were craned and toes were trodden
on as the steamer slackened speed and a line dexterously thrown by a
bluejerseyed deckhand was caught by somebody aboard the yacht the pilot not
insensible to the fact of his being a personage of note carefully divested his
bearded countenance of all expression as he saluted the captain and taking from
the deckstewards obsequiously proffered salver a glass containing fourfingers
of neat bourbon whisky concealed its contents about his person without
perceptible emotion and went up with the first officer upon the upper bridge as
the relieved skipper plunged below the telegraphs clicked their messagethe
leviathan hulk of the liner quivered and began to forge slowly ahead and an
intelligentlooking thinlipped badlyshaved young man in a bowler tweeds and str
```

Tokenisation is splitting a string into words, thus, forming a list of words known as tokens. For tokenisation, we use a library from *nltk* known as *tokenise*. The library consists of function *word_tokenize()* which takes the string, here *T* as the parameter and returns the list as shown in the output.

[26]:
```
# Tokenisation
from nltk.tokenize import word_tokenize

T = word_tokenize(T)
print(T[:1000])
```

```
['\ufeff', 'a', '', '', '', '', 's', '', 'a', '', 's', 'a', 'a', 'a', '',
's', '', 's', '', 's', 'a', '', '', '', '', 'on', 'board', 'the',
'rampatina', 'liner', 'eleven', 'days', 'and', 'a', 'half', 'out', 'from',
'liverpool', 'the', 'usual', 'terrific', 'sensation', 'created', 'by', 'the',
'appearance', 'of', 'the', 'pilotyacht', 'prevailed', 'necks', 'were', 'craned',
'and', 'toes', 'were', 'trodden', 'on', 'as', 'the', 'steamer', 'slackened',
'speed', 'and', 'a', 'line', 'dexterously', 'thrown', 'by', 'a', 'bluejerseyed',
'deckhand', 'was', 'caught', 'by', 'somebody', 'aboard', 'the', 'yacht', 'the',
'pilot', 'not', 'insensible', 'to', 'the', 'fact', 'of', 'his', 'being', 'a',
'personage', 'of', 'note', 'carefully', 'divested', 'his', 'bearded',
'countenance', 'of', 'all', 'expression', 'as', 'he', 'saluted', 'the',
'captain', 'and', 'taking', 'from', 'the', 'decksteward', '', 's',
'obsequiously', 'proffered', 'salver', 'a', 'glass', 'containing',
'fourfingers', 'of', 'neat', 'bourbon', 'whisky', 'concealed', 'its',
'contents', 'about', 'his', 'person', 'without', 'perceptible', 'emotion',
'and', 'went', 'up', 'with', 'the', 'first', 'officer', 'upon', 'the', 'upper',
'bridge', 'as', 'the', 'relieved', 'skipper', 'plunged', 'below', 'the',
'telegraphs', 'clicked', 'their', 'messagethe', 'leviathan', 'hulk', 'of',
'the', 'liner', 'quivered', 'and', 'began', 'to', 'forge', 'slowly', 'ahead',
'and', 'an', 'intelligentlooking', 'thinlipped', 'badlyshaved', 'young', 'man',
'in', 'a', 'bowler', 'tweeds', 'and', 'striped', 'necktie', 'introduced',
'himself', 'to', 'the', 'second', 'officer', 'as', 'an', 'emissary', 'of',
'the', 'press', '', 'mr', 'cyrus', 'k', 'pillson', 'new', 'york', 'yeller',
'pleased', 'to', 'know', 'you', 'sir', '', 'said', 'the', 'second', 'officer',
'', 'step', 'into', 'the', 'smokeroom', 'this', 'way', 'barsteward', 'a',
'brandy', 'cocktail', 'for', 'me', 'and', 'you', 'sir', 'order', 'whatever',
```

'you', 'are', 'most', 'in', 'the', 'habit', 'of', 'hoisting', 'whisky',
'straight', 'now', 'sir', 'happy', 'to', 'afford', 'you', 'what', 'information',
'i', 'can', '', '', 'i', 'presume', '', 'observed', 'the', 'young',
'gentleman', 'of', 'the', 'press', 'settling', 'himself', 'on', 'the',
'springy', 'morocco', 'cushions', 'and', 'accepting', 'the', 'second',
'officer', '', 's', 'polite', 'offer', 'of', 'a', 'green', 'havana', 'of',
'the', 'strongest', 'kind', '', 'that', 'you', 'have', 'had', 'a', 'smooth',
'passage', 'considerin', '', 'the', 'time', 'of', 'year', '', '', 'smooth',
'', 'the', 'second', 'officer', 'carefully', 'reversed', 'in', 'his', 'reply',
'the', 'pressman', '', 's', 'remark', '', 'well', 'yes', 'the', 'time', 'of',
'year', 'considered', 'a', 'smooth', 'passage', 'i', 'take', 'it', 'we', 'have',
'had', '', '', 'no', 'fogs', '', 'interrogated', 'the', 'young', 'gentleman',
'clicking', 'the', 'elastic', 'band', 'of', 'a', 'notebook', 'which',
'projected', 'from', 'his', 'breastpocket', '', 'fogs', 'no', '', 'said',
'the', 'second', 'officer', '', 'you', 'didn', '', 't', 'chance', '',
'pursued', 'the', 'young', 'gentleman', 'of', 'the', 'press', 'taking', 'his',
'short', 'drink', 'from', 'the', 'steward', '', 's', 'salver', 'and',
'throwing', 'it', 'contemptuously', 'down', 'his', 'throat', '', 'to', 'fall',
'in', 'with', 'a', 'berg', 'off', 'the', 'bank', 'did', 'you', '', '', 'not',
'a', 'smell', 'of', 'one', '', 'replied', 'the', 'second', 'officer', 'with',
'decision', '', 'ran', 'into', 'a', 'derelict', 'hencoop', 'perhaps', '',
'persisted', 'the', 'young', 'gentleman', 'concealing', 'the', 'worn', 'sole',
'of', 'a', 'wearied', 'boot', 'from', 'the', 'searching', 'glare', 'of', 'the',
'electric', 'light', 'by', 'tucking', 'it', 'underneath', 'him', '', 'or',
'an', 'old', 'lady', '', 's', 'bonnetbox', 'or', 'a', 'rubber', 'doll', 'some',
'woman', '', 's', 'baby', 'had', 'lost', 'overboard', 'no', '', 'he',
'echoed', 'as', 'the', 'second', 'officer', 'shook', 'his', 'head', '', 'then',
'how', 'in', 'thunder', 'did', 'you', 'manage', 'to', 'lose', 'twenty', 'feet',
'of', 'your', 'portrail', '', '', 'carried', 'away', '', 'said', 'the',
'second', 'officer', 'offering', 'the', 'young', 'press', 'gentleman', 'a',
'light', '', 'no', 'thanks', 'always', 'eat', 'mine', '', 'said', 'the',
'young', 'press', 'gentleman', 'gracefully', '', 'matter', 'of', 'taste', '',
'observed', 'the', 'second', 'officer', 'blowing', 'blue', 'rings', '', 'i',
'guess', 'so', 'and', 'i', '', 've', 'a', 'taste', 'for', 'knowing', 'how',
'you', 'came', '', 'said', 'the', 'young', 'pressman', '', 'to', 'part',
'with', 'that', 'twenty', 'foot', 'of', 'rail', '', '', 'carried', 'away',
'', 'said', 'the', 'second', 'officer', '', 'i', 'kin', 'see', 'that', '',
'retorted', 'the', 'visitor', '', 'it', 'was', 'carried', 'away', '', 'said',
'the', 'second', 'officer', '', 'by', 'an', 'elephant', '', '', 'a', 'pet',
'you', 'had', 'running', 'about', 'aboard', '', 'queried', 'the', 'pressman',
'with', 'imperturbable', 'coolness', '', 'a', 'passenger', '', 'returned',
'the', 'second', 'officer', 'with', 'equal', 'calm', 'there', 'was', 'a',
'snap', 'and', 'the', 'pressman', '', 's', 'notebook', 'was', 'open', 'on',
'his', 'knee', 'the', 'pencil', 'vibrated', 'over', 'the', 'virgin', 'page',
'when', 'a', 'curious', 'utterance', 'between', 'a', 'wail', 'a', 'cough',
'and', 'a', 'roar', 'made', 'the', 'hand', 'that', 'held', 'it', 'start', '',
'yarrrr', 'ohowgh', 'yarr', '', 'the', 'melancholy', 'sound', 'came', 'from',
'without', 'borne', 'on', 'the', 'cool', 'breeze', 'of', 'a', 'late',

```
'afternoon', 'in', 'march', 'through', 'the', 'open', 'ventilators', '',
'might', 'that', '', 'queried', 'the', 'young', 'gentleman', 'of', 'the',
'press', '', 'be', 'an', 'expression', 'of', 'opinion', 'on', 'the', 'part',
'of', 'the', 'elephant', '', '', 'lord', 'love', 'you', 'no', '', 'said',
'the', 'second', 'officer', '', 'it', '', 's', 'the', 'leopard', '', 'he',
'added', 'after', 'a', 'second', '', 's', 'pause', '', 'or', 'the', 'puma',
'', '', 'do', 'you', 'happen', 'to', 'have', 'a', 'menagerie', 'aboard', '',
'inquired', 'the', 'pressman', 'making', 'a', 'note', 'in', 'shorthand', '',
'no', 'sir', 'the', 'beastselephants', 'leopards', 'and', 'a', 'box', 'of',
'cobrasare', 'invoiced', 'from', 'the', 'london', 'docks', 'to', 'a',
'wealthy', 'amateur', 'in', 'new', 'york', 'state', 'not', 'an', 'iron', 'king',
'or', 'a', 'corn', 'king', 'or', 'a', 'cotton', 'king', 'or', 'a', 'pickle',
'king', 'or', 'a', 'kerosene', 'king', '', 'said', 'the', 'second', 'officer',
'with', 'a', 'steady', 'upper', 'lip', '', 'but', 'a', 'chewinggum', 'king',
'', '', 'if', 'you', 'mean', 'shadland', 'c', 'mcoster', '', 'said', 'the',
'pressman', '', 'my', 'mother', 'is', 'his', 'cousin', 'they', 'used', 'to',
'chew', 'gum', 'together', 'in', 'school', 'recess', 'sir', 'little',
'guessing', 'that', 'shad', 'would', 'one', 'day', 'soar', 'on', 'wings',
'made', 'of', 'that', 'article', 'to', 'the', 'realms', 'of', 'gilded',
'plutocracy', '', '', 'i', 'rather', 'imagine', 'the', 'name', 'you',
'mention', 'to', 'be', 'the', 'right', 'one', '', 'said', 'the', 'second',
'officer', 'cautiously', '', 'but', 'i', 'won', '', 't', 'commit', 'myself',
'the', 'beasts', 'shipped', 'from', 'liverpool', 'are', 'intended', 'as', 'a',
'present', 'for', 'the', 'purchaser', '', 's', 'infant', 'daughter', 'on',
'her', 'fifth', 'birthday', '', '', 'yarrrr', 'ohowgh', 'ohowgh', '',
'again', 'the', 'coughing', 'roar', 'vibrated', 'through', 'the', 'smokeroom',
'then', 'the', 'chorus', 'of', '', 'hail', 'columbia', '', 'rose', 'from',
'the', 'promenade', 'deck', 'where', 'the', 'lady', 'passengers', 'were',
'assembled', 'ready', 'to', 'wave', 'starred', 'and', 'striped', 'silk',
'pockethandkerchiefs', 'and', 'exchange', 'patriotic', 'sentiments', 'at',
'the', 'first', 'glimpse', 'of', 'land', '', 'it', '', 's', 'not', 'what',
'i', 'should', 'call', 'a', 'humly', 'voice', 'that', 'of', 'the', 'leopard',
'', 'observed', 'the', 'pressman', 'controlling', 'a', 'slight', 'shiver', '',
'children', 'have']
```

Lemmatization is grouping together of the different forms of a word having similar meaning that is reducing the inflected forms and sometimes derivationally related forms of a word to a common base form to ease the analysis of the document.

[27]:
```
# Lemmatisation
lemmatizer = WordNetLemmatizer()

# forming a set of stopwords from english language
stop_words = set(stopwords.words('english'))

# creating an empty list
lemmatized_T=[]
```

```
# traversing all the words
for word in T:
    # if the word has a length greater than or equal to 2 and is not a stopword
    if len(word) >= 2 and word not in stop_words:
        # then we append the word into the list lemmatized_T after performing␣
 ↪lemmatization
        # using the lemmatize() function
        lemmatized_T.append(lemmatizer.lemmatize(word))

# printing the list of lemmatized words
print(lemmatized_T[:1000])
```

```
['board', 'rampatina', 'liner', 'eleven', 'day', 'half', 'liverpool', 'usual',
'terrific', 'sensation', 'created', 'appearance', 'pilotyacht', 'prevailed',
'neck', 'craned', 'toe', 'trodden', 'steamer', 'slackened', 'speed', 'line',
'dexterously', 'thrown', 'bluejerseyed', 'deckhand', 'caught', 'somebody',
'aboard', 'yacht', 'pilot', 'insensible', 'fact', 'personage', 'note',
'carefully', 'divested', 'bearded', 'countenance', 'expression', 'saluted',
'captain', 'taking', 'decksteward', 'obsequiously', 'proffered', 'salver',
'glass', 'containing', 'fourfingers', 'neat', 'bourbon', 'whisky', 'concealed',
'content', 'person', 'without', 'perceptible', 'emotion', 'went', 'first',
'officer', 'upon', 'upper', 'bridge', 'relieved', 'skipper', 'plunged',
'telegraph', 'clicked', 'messagethe', 'leviathan', 'hulk', 'liner', 'quivered',
'began', 'forge', 'slowly', 'ahead', 'intelligentlooking', 'thinlipped',
'badlyshaved', 'young', 'man', 'bowler', 'tweed', 'striped', 'necktie',
'introduced', 'second', 'officer', 'emissary', 'press', 'mr', 'cyrus',
'pillson', 'new', 'york', 'yeller', 'pleased', 'know', 'sir', 'said', 'second',
'officer', 'step', 'smokeroom', 'way', 'barsteward', 'brandy', 'cocktail',
'sir', 'order', 'whatever', 'habit', 'hoisting', 'whisky', 'straight', 'sir',
'happy', 'afford', 'information', 'presume', 'observed', 'young', 'gentleman',
'press', 'settling', 'springy', 'morocco', 'cushion', 'accepting', 'second',
'officer', 'polite', 'offer', 'green', 'havana', 'strongest', 'kind', 'smooth',
'passage', 'considerin', 'time', 'year', 'smooth', 'second', 'officer',
'carefully', 'reversed', 'reply', 'pressman', 'remark', 'well', 'yes', 'time',
'year', 'considered', 'smooth', 'passage', 'take', 'fog', 'interrogated',
'young', 'gentleman', 'clicking', 'elastic', 'band', 'notebook', 'projected',
'breastpocket', 'fog', 'said', 'second', 'officer', 'chance', 'pursued',
'young', 'gentleman', 'press', 'taking', 'short', 'drink', 'steward', 'salver',
'throwing', 'contemptuously', 'throat', 'fall', 'berg', 'bank', 'smell', 'one',
'replied', 'second', 'officer', 'decision', 'ran', 'derelict', 'hencoop',
'perhaps', 'persisted', 'young', 'gentleman', 'concealing', 'worn', 'sole',
'wearied', 'boot', 'searching', 'glare', 'electric', 'light', 'tucking',
'underneath', 'old', 'lady', 'bonnetbox', 'rubber', 'doll', 'woman', 'baby',
'lost', 'overboard', 'echoed', 'second', 'officer', 'shook', 'head', 'thunder',
'manage', 'lose', 'twenty', 'foot', 'portrail', 'carried', 'away', 'said',
'second', 'officer', 'offering', 'young', 'press', 'gentleman', 'light',
'thanks', 'always', 'eat', 'mine', 'said', 'young', 'press', 'gentleman',
```

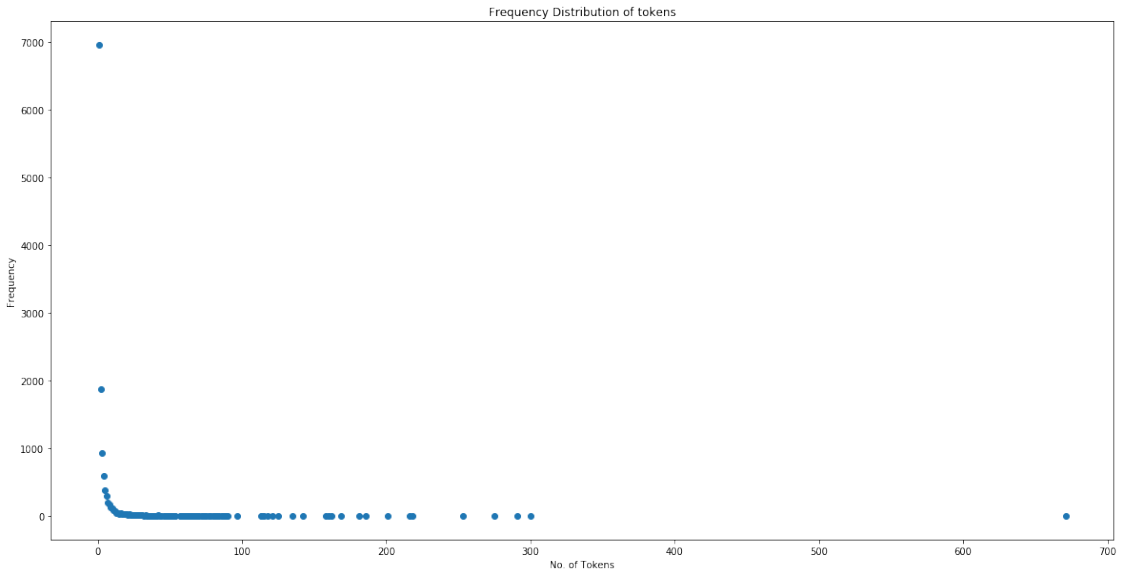
'gracefully', 'matter', 'taste', 'observed', 'second', 'officer', 'blowing',
'blue', 'ring', 'guess', 'taste', 'knowing', 'came', 'said', 'young',
'pressman', 'part', 'twenty', 'foot', 'rail', 'carried', 'away', 'said',
'second', 'officer', 'kin', 'see', 'retorted', 'visitor', 'carried', 'away',
'said', 'second', 'officer', 'elephant', 'pet', 'running', 'aboard', 'queried',
'pressman', 'imperturbable', 'coolness', 'passenger', 'returned', 'second',
'officer', 'equal', 'calm', 'snap', 'pressman', 'notebook', 'open', 'knee',
'pencil', 'vibrated', 'virgin', 'page', 'curious', 'utterance', 'wail', 'cough',
'roar', 'made', 'hand', 'held', 'start', 'yarrrr', 'ohowgh', 'yarr',
'melancholy', 'sound', 'came', 'without', 'borne', 'cool', 'breeze', 'late',
'afternoon', 'march', 'open', 'ventilator', 'might', 'queried', 'young',
'gentleman', 'press', 'expression', 'opinion', 'part', 'elephant', 'lord',
'love', 'said', 'second', 'officer', 'leopard', 'added', 'second', 'pause',
'puma', 'happen', 'menagerie', 'aboard', 'inquired', 'pressman', 'making',
'note', 'shorthand', 'sir', 'beastselephants', 'leopard', 'box', 'cobrasare',
'invoiced', 'london', 'dock', 'wealthy', 'amateur', 'new', 'york', 'state',
'iron', 'king', 'corn', 'king', 'cotton', 'king', 'pickle', 'king', 'kerosene',
'king', 'said', 'second', 'officer', 'steady', 'upper', 'lip', 'chewinggum',
'king', 'mean', 'shadland', 'mcoster', 'said', 'pressman', 'mother', 'cousin',
'used', 'chew', 'gum', 'together', 'school', 'recess', 'sir', 'little',
'guessing', 'shad', 'would', 'one', 'day', 'soar', 'wing', 'made', 'article',
'realm', 'gilded', 'plutocracy', 'rather', 'imagine', 'name', 'mention',
'right', 'one', 'said', 'second', 'officer', 'cautiously', 'commit', 'beast',
'shipped', 'liverpool', 'intended', 'present', 'purchaser', 'infant',
'daughter', 'fifth', 'birthday', 'yarrrr', 'ohowgh', 'ohowgh', 'coughing',
'roar', 'vibrated', 'smokeroom', 'chorus', 'hail', 'columbia', 'rose',
'promenade', 'deck', 'lady', 'passenger', 'assembled', 'ready', 'wave',
'starred', 'striped', 'silk', 'pockethandkerchiefs', 'exchange', 'patriotic',
'sentiment', 'first', 'glimpse', 'land', 'call', 'humly', 'voice', 'leopard',
'observed', 'pressman', 'controlling', 'slight', 'shiver', 'child', 'queer',
'taste', 'said', 'second', 'officer', 'well', 'old', 'spot', 'lively', 'bingo',
'dead', 'bingo', 'queried', 'pressman', 'bingo', 'elephant', 'said', 'second',
'officer', 'passing', 'palm', 'brown', 'right', 'hand', 'upper', 'lip',
'pressman', 'made', 'rapid', 'note', 'particular', 'deathbed', 'scene',
'likely', 'interest', 'youwhy', 'welcome', 'em', 'white', 'said', 'pressman',
'warmly', 'licking', 'pencil', 'elephant', 'die', 'seasickness', 'said',
'second', 'officer', 'calmly', 'seen', 'thing', 'worth', 'seeingmyself',
'said', 'pressman', 'enviously', 'seasick', 'elephant', 'professional',
'ladynurse', 'attendance', 'said', 'second', 'officer', 'complete', 'stem',
'stern', 'print', 'gown', 'white', 'apron', 'flyaway', 'caprigging', 'ward',
'shoe', 'pressman', 'grunted', 'lack', 'interest', 'doubled', 'corner',
'smokeroom', 'divan', 'notebook', 'balanced', 'bulging', 'shirtfront', 'made',
'furious', 'note', 'second', 'officer', 'waited', 'pencil', 'seemed', 'hungry',
'fed', 'little', 'information', 'girl', 'came', 'aboard', 'liverpool',
'mackintosh', 'holdall', 'little', 'black', 'shiny', 'bag', 'went', 'noticed',
'passing', 'sort', 'way', 'freshcolored', 'tidylooking', 'young', 'woman',
'rather', 'plump', 'bow', 'air', 'though', 'meant', 'get', 'full', 'money',
'worth', 'elevenpound', 'fare', 'cheap', 'tariff', 'filled', 'passengerlists',

'fairly', 'full', 'long', 'score', 'thing', 'attend', 'special', 'derrick',
'rigged', 'sling', 'elephant', 'cage', 'aboard', 'capital', 'one', 'sound',
'indian', 'teak', 'strengthened', 'steelmust', 'cost', 'mint', 'money',
'stowed', 'lot', 'sweat', 'swearing', 'promenade', 'deck', 'abaft', 'funnel',
'bolting', 'ring', 'specially', 'screwed', 'deck', 'passing', 'wire', 'hawser',
'across', 'top', 'made', 'fast', 'port', 'starboard', 'davit', 'rigging',
'weatherscreens', 'double', 'tarpaulin', 'keep', 'bingo', 'warm', 'dry',
'beast', 'shipped', 'lee', 'forward', 'cabin', 'skylight', 'got', 'job',
'quiet', 'ladylike', 'voice', 'elbow', 'say', 'please', 'officer', 'regard',
'patient', 'wish', 'know', 'ask', 'purser', 'said', 'rather', 'snappishly',
'hot', 'worried', 'headstewardess', 'asked', 'say', 'voice', 'calm',
'determined', 'way', 'referred', 'well', 'say', 'mistake', 'say', 'young',
'ladyfor', 'young', 'lady', 'hospital', 'nurse', 'besides', 'neatly', 'rigged',
'usual', 'uniform', 'mistake', 'allotted', 'bedroom', 'groundfloor', 'far',
'patient', 'possibly', 'hear', 'call', 'night', 'went', 'breeze', 'played',
'white', 'silk', 'bonnetstrings', 'wavy', 'little', 'kink', 'soft', 'brown',
'hair', 'framed', 'forehead', 'want', 'move', 'upper', 'floor', 'mean',
'promenade', 'deck', 'madam', 'say', 'smoothing', 'grin', 'though', 'well',
'enough', 'used', 'odd', 'bungle', 'landfolks', 'make', 'name', 'thing', 'sea',
'flying', 'pencil', 'stopped', 'pressman', 'looked', 'turning', 'shortened',
'cigar', 'teeth', 'come', 'elephant', 'asked', 'said', 'second', 'officer',
'mean', 'promenade', 'deck', 'say', 'patient', 'occupy', 'one', 'cabin', 'port',
'starboard', 'side', 'may', 'ask', 'number', 'name', 'smiled', 'brightly',
'eye', 'teeth', 'making', 'sort', 'flash', 'together', 'cabin', 'say', 'sleep',
'cage', 'patient', 'bingo', 'elephant', 'great', 'pierpont', 'morgan',
'ejaculated', 'pressman', 'previously', 'flying', 'pencil', 'became', 'almost',
'invisible', 'extreme', 'rapidity', 'plied', 'drop', 'perspiration', 'broke',
'upon', 'sallow', 'forehead', 'glory', 'cried', 'another', 'man', 'thought',
'worth', 'run', 'tackle', 'wallowing', 'old', 'tub', 'touched', 'cap', 'went',
'second', 'officer', 'keeping', 'professionally', 'could', 'surprise', 'felt',
'understand', 'madam', 'asked', 'elephant', 'nurse', 'nodded', 'another',
'bright', 'smile', 'told', 'nurse', 'amy', 'st', 'baalam', 'nursing',
'association', 'london', 'specially', 'engaged', 'american', 'gentleman',
'bought', 'elephant', 'shadland', 'mcoster', 'prompted', 'pressman',
'without', 'looking', 'attend', 'animal', 'voyage', 'understood', 'principal',
'patient', 'condition', 'permitted', 'nurse', 'amy', 'pay', 'leopard',
'attention', 'capable', 'appreciating', 'pressure', 'point', 'ohowgh',
'coughed', 'voice', 'outside', 'yarr', 'ohowgh', 'smell', 'land', 'guess',
'said', 'pressman', 'nigger', 'suggested', 'second', 'officer', 'ought',
'heard', 'bingo', 'three', 'day', 'mersey', 'fair', 'wind', 'smooth', 'sea',
'first', 'nothing', 'delighted', 'lady', 'child', 'board', 'like', 'feeding',
'apple', 'nut', 'biscuit', 'thing', 'prigged', 'saloon', 'table', 'seaair',
'must', 'sharpened', 'beast', 'appetite', 'suppose', 'old', 'trunk', 'snorking',
'round', 'day', 'purser', 'naturally', 'wild', 'said', 'must', 'put', 'away',
'hogshead', 'good', 'thing', 'addition', 'allowance', 'hay', 'bread',
'beetroot', 'grain', 'cabbage', 'sugar', 'ca', 'temper', 'asked', 'pressman',
'mild', 'milk', 'kind', 'beast', 'ever', 'breathed', 'elephant', 'lot',
'breathing', 'said', 'second', 'officer', 'lady', 'gentleman', 'upperdeck',

```
'cabin', 'used', 'complain', 'snoring', 'night', 'nurse', 'amy', 'said',
'people', 'complain', 'anything', 'em', 'like', 'smell', 'elephantwhich',
'allow', 'happened']
```

We plotted the relation between Tokens and Frquency as the number of tokens occuring 'f' times(frequency). We didn't plotted the tokens because the tokens name was not visible as there was a huge amount of them.

[28]:
```
# Evaluating frequency distribution of tokens
# The nltk library function FreqDist() returns a dictionary containing key␣
 ↪value pairs where values are the frequency of
# the keys. Here keys are the Tokens.
freq_dist = nltk.FreqDist(lemmatized_T)

# create a dictionary
word_dic = {}

# find the k words occuring f number of times and store in the dictionary
for i in freq_dist.keys():
    if freq_dist[i] not in word_dic:
        word_dic[freq_dist[i]] = 1
    else:
        word_dic[freq_dist[i]] += 1

# plotting a scatter plot diagram of the frequency distribution and tokens
plt.figure(figsize=(20,10))
plt.scatter(word_dic.keys(),word_dic.values())
plt.xlabel("No. of Tokens")
plt.ylabel("Frequency")
plt.title("Frequency Distribution of tokens")
plt.show()
```

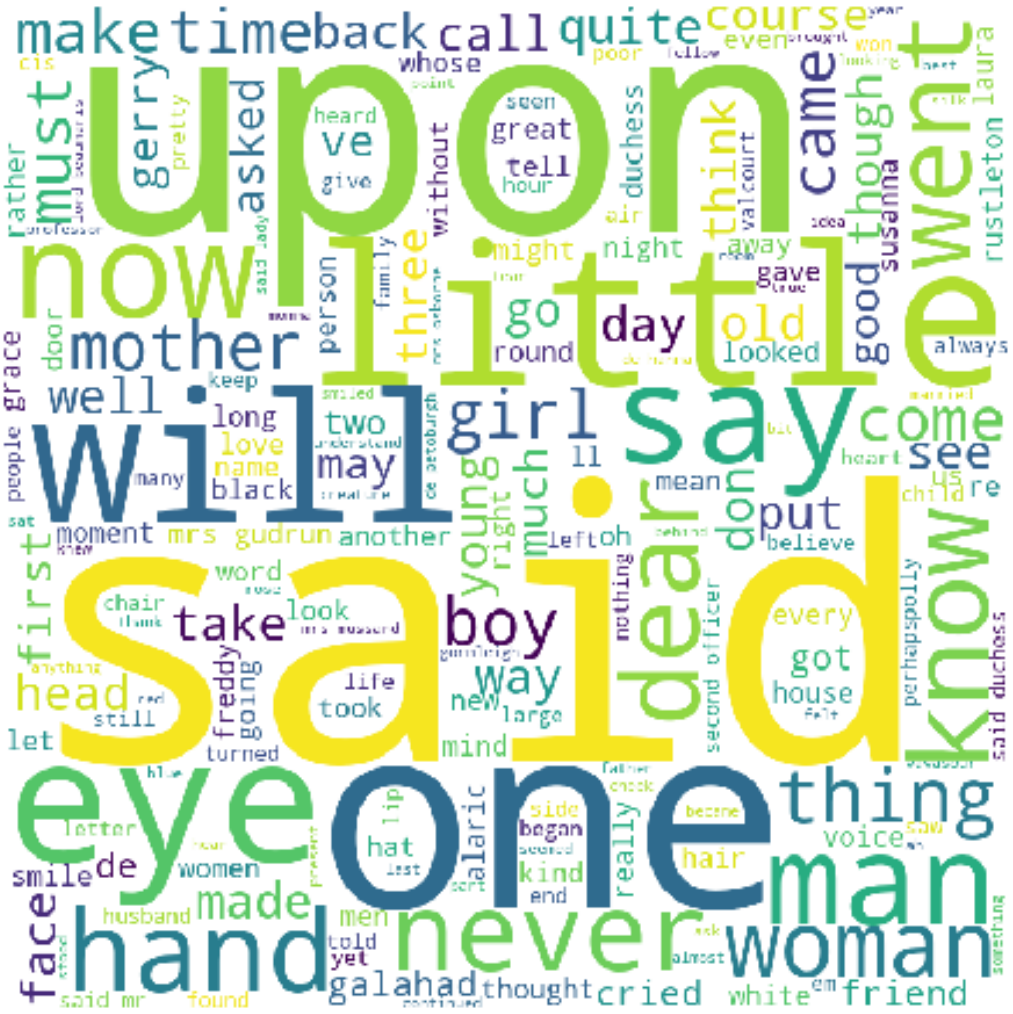


WordCloud is a representation of the textual data according to their frequency. It basically emphasizes the keywords used in the text.

[29]:
```
# WordCloud without removing Stopwords
# We used WORDCLOUD package to create a wordcloud. The function WordCloud takes␣
 ↪in the size and color of the word cloud
# and then returns a worldcloud object based on the frequency of words.

wordcloud = WordCloud(width = 800, height=800,
                      background_color='white',
                      min_font_size=10).generate(wordcloudT)

plt.figure(figsize=(8,8),facecolor=None)
plt.imshow(wordcloud)
plt.axis("off")

plt.show()
```

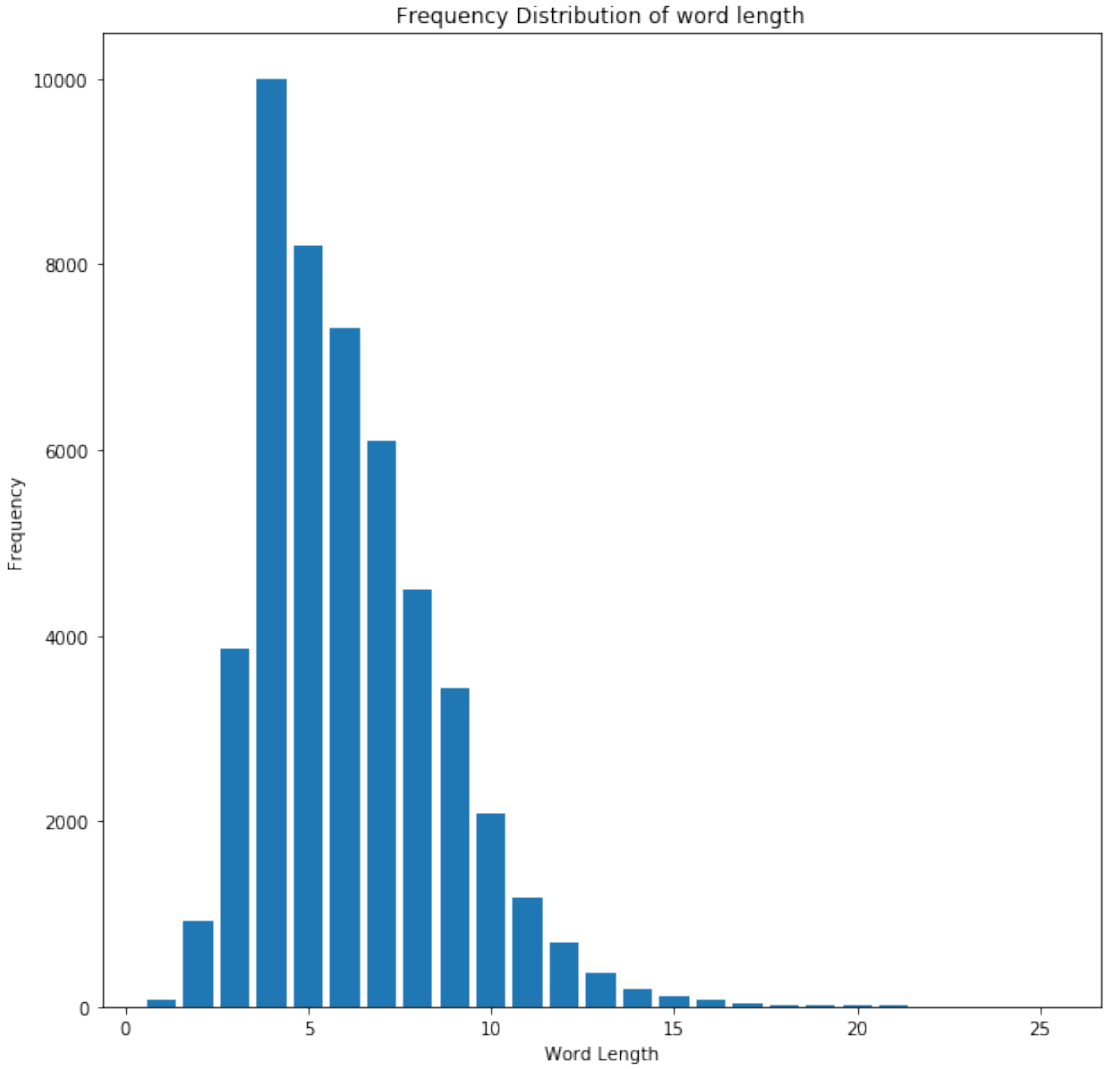
Removing stopwords decreases noise in the data and so other words get more emphasis in the wordcloud.

[30]:
```
# WordCloud after removing Stopwords
wordcloud = WordCloud(width = 800, height=800,
                     background_color='white',
                      stopwords = set(STOPWORDS),
                     min_font_size=10).generate(wordcloudT)

plt.figure(figsize=(8,8),facecolor=None)
plt.imshow(wordcloud)
plt.axis("off")

plt.show()
```



[31]:
```
#Frequency distribution of Word Length
#Result: As the word length increases the frequency of that word decreases. So␣
 ↪large length word occurs less in the book as
#        compared to small length words.
len_list={}

for i in range(len(lemmatized_T)):

    if len(lemmatized_T[i]) not in len_list:
        len_list[len(lemmatized_T[i])] = 1

    else:
        len_list[len(lemmatized_T[i])] += 1
```

```
keys = list(len_list.keys())
values = list(len_list.values())

plt.figure(figsize=(10,10))
plt.bar(keys,values)
plt.xlabel("Word Length")
plt.ylabel("Frequency")
plt.title("Frequency Distribution of word length")
plt.show()
```



Using brown corpus for POS tagging. We used nltk again for this. The tagset we used is universal so the number of tags will be less because in universal tagset, many tags are clustered into one tag and then the pos tagging is performed.

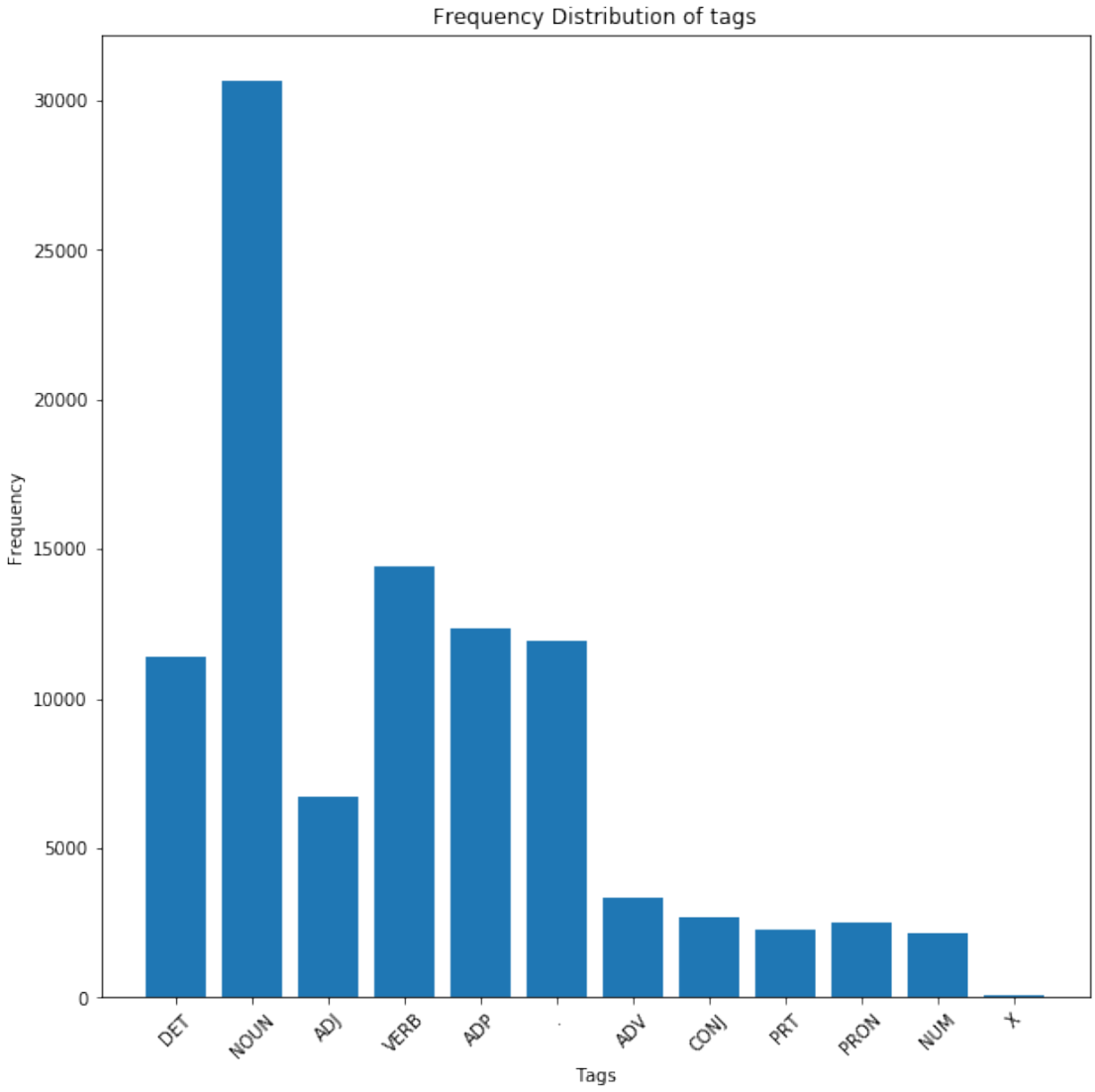
[32]:
```
brown_news_words = brown.tagged_words(categories='news', tagset='universal')
brown_news_words
```

[32]:
```
[('The', 'DET'), ('Fulton', 'NOUN'), ...]
```

[33]:
```
# Frequency distribution of POS tags
fdistw = nltk.FreqDist([t for (w, t) in brown_news_words])
fdistw
```

[33]:
```
FreqDist({'NOUN': 30654, 'VERB': 14399, 'ADP': 12355, '.': 11928, 'DET': 11389,
'ADJ': 6706, 'ADV': 3349, 'CONJ': 2717, 'PRON': 2535, 'PRT': 2264, ...})
```

[34]:
```
# Representation of frequrncy distribution of POS tags using brown corpus
# creating a list of the tokens
keys = list(fdistw.keys())

# creating a list of the frequency of the various tokens
values = list(fdistw.values())

# plotting a bar plot diagram of the frequency distribution
plt.figure(figsize=(10,10))
plt.bar(keys,values)
plt.xlabel("Tags")
plt.xticks(rotation=45)
plt.ylabel("Frequency")
plt.title("Frequency Distribution of tags")
plt.show()
```


Frequency Distribution of tags
Frequency
30000
25000
20000
15000
10000
5000
0
DET
NOUN
ADJ
VERB
ADP
.
ADV
CONJ
PRT
PRON
NUM
X
Tags

[ ]: