# Computational Linguistics

## Summer 2021, IIIT Hyderabad

## Project 1 – Wordcloud Generation

### Algorithm

The first section of the code scrapes the Internet for 10,000 sentences. The websites are HTML versions of four classic English books: Alice in Wonderland, Jungle Book, The Wizard of Oz and The Adventures of Tom Sawyer.

The raw data is written to a file `raw.txt` in the same directory, and the user is prompted to enter a filename to read the new raw data. If the user wishes to use the same data, `raw.txt` should be given as input. The file is then read.

The raw data is then

- tokenised into words and then sentences
- cleaned (rid of punctuation, abbreviations, symbols and foreign words)
- rid of stopwords
- PoS-tagged
- stemmed and lemmatised

The NLTK library is used for all the above steps, as well as for graphing. The BeautifulSoup library is used for scraping.

Thereafter, the frequency graphs are created with the top 20 tokens at each step (seven graphs; see below). Then the wordcloud is generated with the top 50 lemmata. Note that a new wordcloud will be generated every time the code is run.

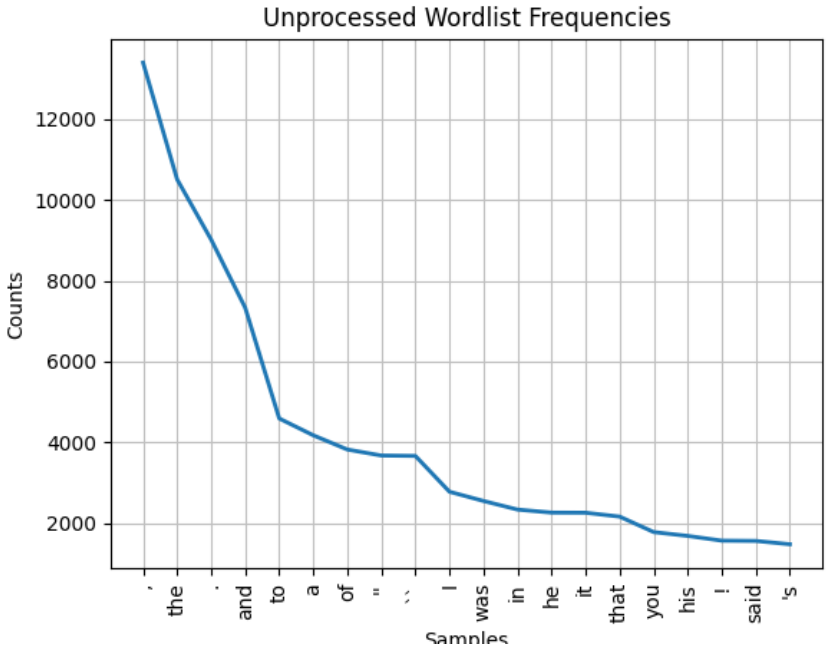
These numbers were chosen taking into a account a reasonable size and shape of the graphs. More than 50 words do not fit in a 2000 x 3000 px image without getting very cluttered, and 50 words is sufficient to give an idea of the contents of the text. Furthermore, a graph with more than 20 words gets too big, and 20 words are enough to judge the correctness of the method.

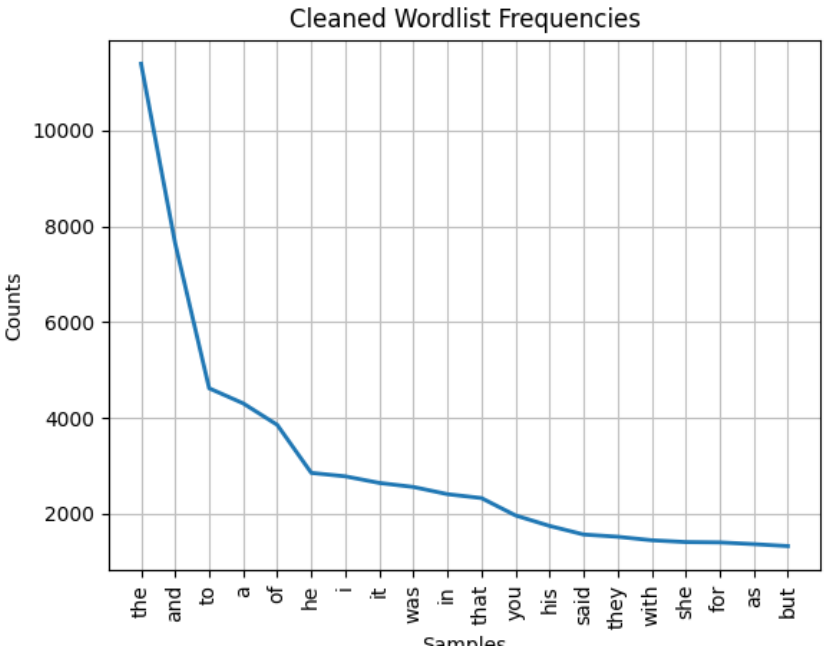
### Analysis

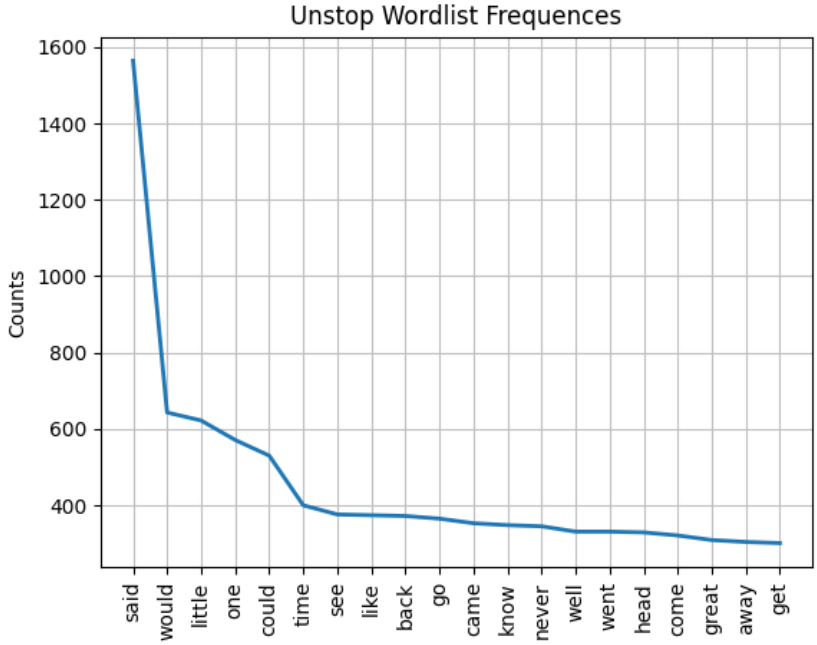
- The forms of all graphs of tokens were roughly similar, displaying a steep drop from the beginning and a flattening towards the end.
- In the unprocessed wordlist graph, the most frequent tokens were punctuation, which occurred almost once every sentence. This explains the need to clean the data by removing such symbols, as they carry no real meaning.
- The top 20 words in the cleaned wordlist were exclusively stopwords, and convey no idea as to the content of the data. This, too, agrees with our idea of stopwords as generally being the most frequent words.
- The differences between the "Raw Taglist" graph and the "Unstop Taglist" graph is mainly in the frequencies of the tags `DT`, `PRP`, `VBZ` and `VBD`,

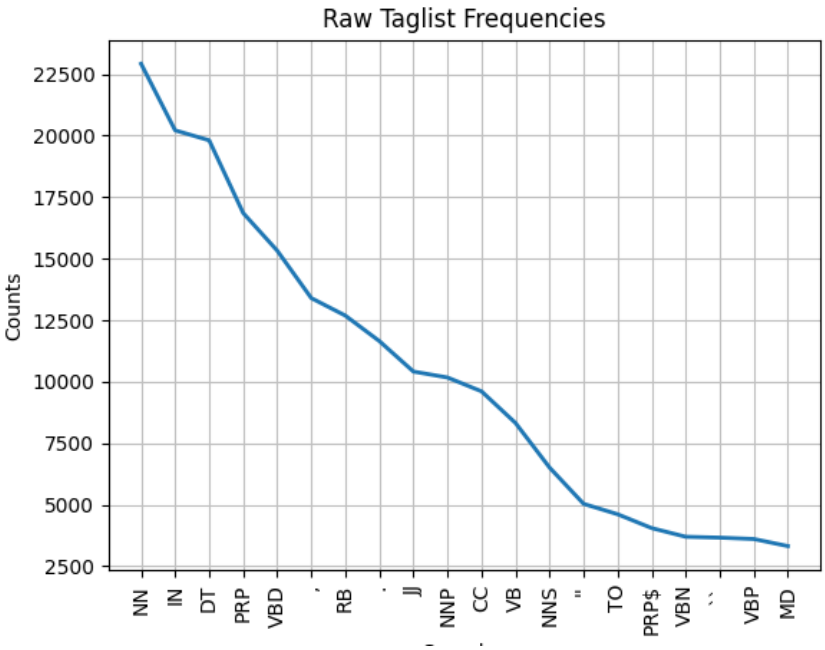
suggesting that most stopwords belong to these parts of speech. This is in accordance with our intuition.

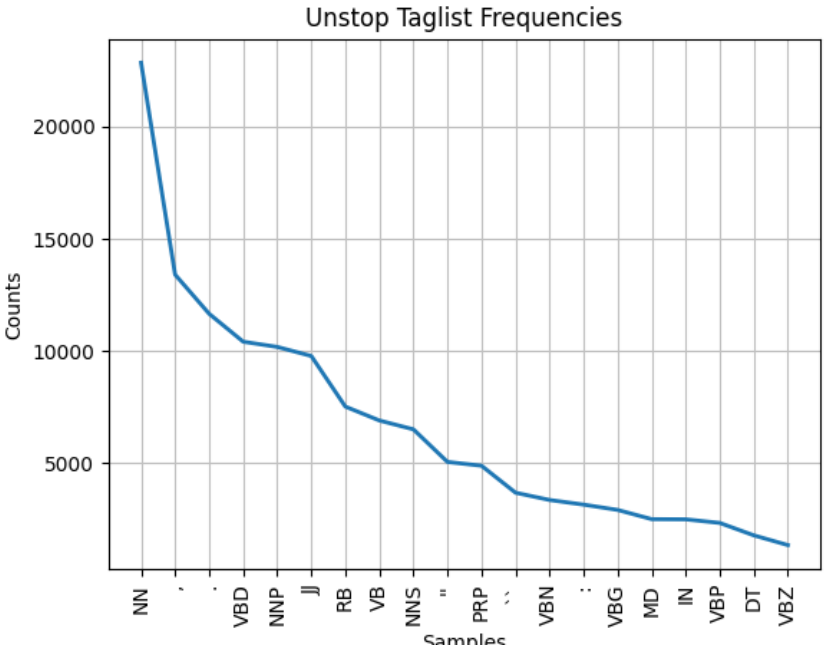
- After removing stopwords, the most frequent tags are those of nouns, verbs and adjectives, which are usually content words. Therefore we expect that the top 20 words with stopwords removed reflect the content of the data.
- This is in fact what we see in the three graphs of unstop words, stems and lemmata. Although there are no duplicate wordforms in the top 20 of the unstop list, if there were, the lemmata wordlist would take care of it; we can use either in the present case.

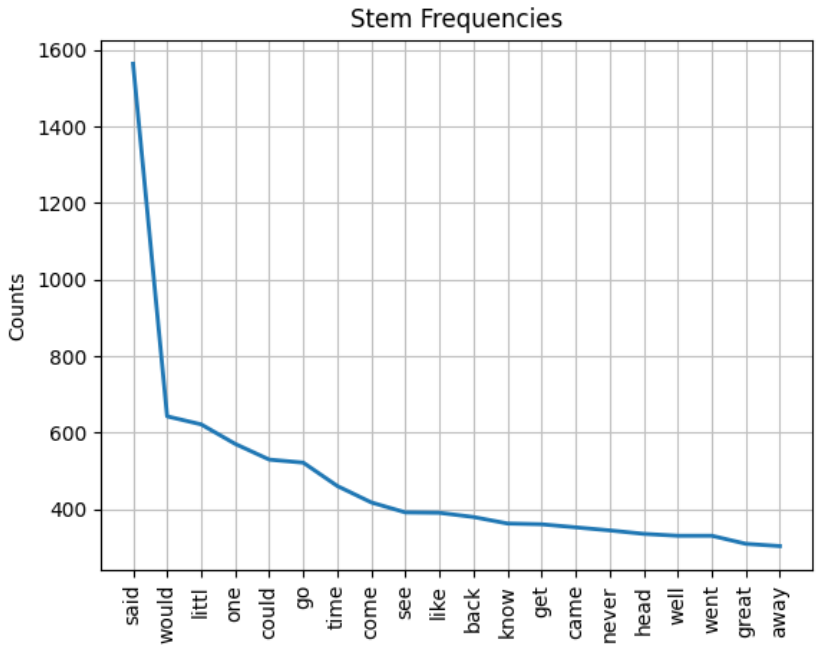


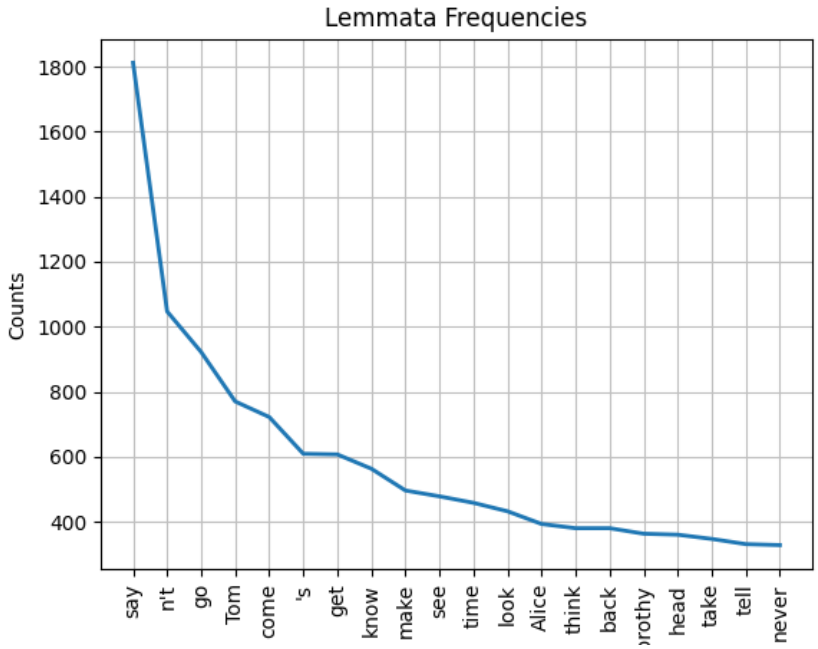

Cleaned Wordlist Frequencies
Counts
10000
8000
6000
4000
2000
the
and
to
a
of
he
i
it
was
in
that
you
his
said
they
with
she
for
as
but


Unstop Wordlist Frequences
Counts
1600
1400
1200
1000
800
600
400
said
would
little
one
could
time
see
like
back
go
came
know
never
well
went
head
come
great
away
get


Raw Taglist Frequencies
Counts
22500
20000
17500
15000
12500
10000
7500
5000
2500
NN
IN
DT
PRP
VBD
,
RB
.
JJ
NNP
CC
VB
NNS
"
TO
PRP$
VBN
``
VBP
MD


Unstop Taglist Frequencies
Counts
20000
15000
10000
5000
NN
,
.
VBD
NNP
JJ
RB
VB
NNS
"
PRP
``
VBN
:
VBG
MD
IN
VBP
DT
VBZ
Samples


Stem Frequencies
Counts
1600
1400
1200
1000
800
600
400
said
would
littl
one
could
go
time
come
see
like
back
know
get
came
never
head
well
went
great
away


Lemmata Frequencies
Counts
1800
1600
1400
1200
1000
800
600
400
say
n't
go
Tom
come
's
get
know
make
see
time
look
Alice
think
back
head
take
tell
never