leni_data

April 10, 2022

1 Data Preparation

[1]:
```
!export PATH=/Library/TeX/texbin:$PATH
import warnings
warnings.filterwarnings("ignore")
```

```
'export' is not recognized as an internal or external command,
operable program or batch file.
```

[2]:
```
# Read the tweets from CSV
import pandas as pd
tweets = pd.read_csv('tweets_leni.csv', index_col=False)
tweets.head()
```

[2]:

	id	handle	created_at	text	cleaned_text	supercleaned_text
0	1461645389619027971	gonzoyyy	2021-11-19 10:39:51+00:00	i will never apologize for being politically a…	i will never apologize for being politically a…	never apologize politically active social medi…
1	1461645317158227969	charliekogure	2021-11-19 10:39:34+00:00	@just_ice1992 @marortoll kababuyan from them i…	kababuyan from them is the new pandemic if Len…	
2	1461645058688487429	beetookkk	2021-11-19 10:38:32+00:00	@manigsaca @kweenCelestia Leni has been irrele…	Leni has been irrelevant since 2016.	
3	1461644745134907396	adrianayalin	2021-11-19 10:37:17+00:00	VP Leni thanked medical frontliners of Quezon …	VP Leni thanked medical frontliners of Quezon …	
4	1461644676876828676	manigsaca	2021-11-19 10:37:01+00:00	@kweenCelestia LENI is becoming like Pacquiao…	LENI is becoming like Pacquiao.. irrelevant	

```
1                    kababuyan new pandemic leni wins
2                          leni irrelevant since 2016
3  vp leni thanked medical frontliners quezon con…
4              leni becoming like pacquiao irrelevant
```

[3]:
```
!pip install spacy
import warnings
import spacy, statistics
from spacy.tokens import Doc
from nltk.corpus import sentiwordnet as swn
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

nlp = spacy.load("en_core_web_sm")
sia = SentimentIntensityAnalyzer()
```

```
Requirement already satisfied: spacy in d:\anaconda3\lib\site-packages (3.2.4)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in
d:\anaconda3\lib\site-packages (from spacy) (1.0.6)
Requirement already satisfied: typer<0.5.0,>=0.3.0 in d:\anaconda3\lib\site-
packages (from spacy) (0.4.1)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in d:\anaconda3\lib\site-
packages (from spacy) (2.0.7)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in d:\anaconda3\lib\site-
packages (from spacy) (2.0.6)
Requirement already satisfied: blis<0.8.0,>=0.4.0 in d:\anaconda3\lib\site-
packages (from spacy) (0.7.7)
Requirement already satisfied: pathy>=0.3.5 in d:\anaconda3\lib\site-packages
(from spacy) (0.6.1)
Requirement already satisfied: jinja2 in d:\anaconda3\lib\site-packages (from
spacy) (2.11.3)
Requirement already satisfied: click<8.1.0 in d:\anaconda3\lib\site-packages
(from spacy) (8.0.3)
Requirement already satisfied: packaging>=20.0 in d:\anaconda3\lib\site-packages
(from spacy) (21.0)
Requirement already satisfied: thinc<8.1.0,>=8.0.12 in d:\anaconda3\lib\site-
packages (from spacy) (8.0.15)
Requirement already satisfied: numpy>=1.15.0 in d:\anaconda3\lib\site-packages
(from spacy) (1.20.3)
Requirement already satisfied: srsly<3.0.0,>=2.4.1 in d:\anaconda3\lib\site-
packages (from spacy) (2.4.2)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.8 in
d:\anaconda3\lib\site-packages (from spacy) (3.0.9)
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<1.9.0,>=1.7.4 in
d:\anaconda3\lib\site-packages (from spacy) (1.8.2)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in d:\anaconda3\lib\site-
packages (from spacy) (3.3.0)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in d:\anaconda3\lib\site-
```

```
packages (from spacy) (2.27.1)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in
d:\anaconda3\lib\site-packages (from spacy) (1.0.2)
Requirement already satisfied: setuptools in d:\anaconda3\lib\site-packages
(from spacy) (58.0.4)
Requirement already satisfied: wasabi<1.1.0,>=0.8.1 in d:\anaconda3\lib\site-
packages (from spacy) (0.9.1)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in d:\anaconda3\lib\site-
packages (from spacy) (3.0.6)
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in d:\anaconda3\lib\site-
packages (from spacy) (4.62.3)
Requirement already satisfied: colorama in d:\anaconda3\lib\site-packages (from
click<8.1.0->spacy) (0.4.4)
Requirement already satisfied: pyparsing>=2.0.2 in d:\anaconda3\lib\site-
packages (from packaging>=20.0->spacy) (3.0.4)
Requirement already satisfied: smart-open<6.0.0,>=5.0.0 in
d:\anaconda3\lib\site-packages (from pathy>=0.3.5->spacy) (5.2.1)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
d:\anaconda3\lib\site-packages (from
pydantic!=1.8,!=1.8.1,<1.9.0,>=1.7.4->spacy) (3.10.0.2)
Requirement already satisfied: idna<4,>=2.5 in d:\anaconda3\lib\site-packages
(from requests<3.0.0,>=2.13.0->spacy) (3.2)
Requirement already satisfied: charset-normalizer~=2.0.0 in
d:\anaconda3\lib\site-packages (from requests<3.0.0,>=2.13.0->spacy) (2.0.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in d:\anaconda3\lib\site-
packages (from requests<3.0.0,>=2.13.0->spacy) (1.26.7)
Requirement already satisfied: certifi>=2017.4.17 in d:\anaconda3\lib\site-
packages (from requests<3.0.0,>=2.13.0->spacy) (2021.10.8)
Requirement already satisfied: MarkupSafe>=0.23 in d:\anaconda3\lib\site-
packages (from jinja2->spacy) (1.1.1)
```

[4]:
```
noun_subject_search = lambda doc: [token.lower_ for token in doc if token.dep_␣
↪== 'nsubj' and token.is_alpha]

robredo_subject_search = lambda doc: True if 'leni' in [
    token.lower_ for token in doc
    if token.dep_ == 'nsubj'
    and token.is_alpha
] else False
```

[5]:
```
# Define pipeline extensions

# Search for all lemmatized verbs and adjectives in each document
verb_search = lambda doc: [token.lemma_ for token in doc if token.pos_ ==␣
↪'VERB' and token.is_alpha]
adjective_search = lambda doc: [token.lemma_ for token in doc if token.pos_ ==␣
↪'ADJ' and token.is_alpha]
```

```
# Apply word-level sentiment analysis to each verb and adjective in each␣
↪document
def word_level_sentiment_analysis(tokens):
    sentiments = {}
    for t in tokens:
        synset = list(swn.senti_synsets(t))

        # Generate sentiment polarities if there are any synsets
        if synset:
            sent = list(swn.senti_synsets(t))[0]
            sentiments[t] = {
                'pos': sent.pos_score(),
                'neg': sent.neg_score(),
                'compound': (sent.pos_score() - sent.neg_score())/2
            }
        # Generate blank if no synsets are found
        else:
            sentiments[t] = {}

    # Get the average compound sentiment for every token found
    if sentiments:
        avg_sentiments = [v['compound'] for (k, v) in sentiments.items() if v]
        if avg_sentiments:
            sentiments['average_sentiment'] = statistics.mean(avg_sentiments)
        else:
            sentiments['average_sentiment'] = 0
    else:
        sentiments['average_sentiment'] = 0

    return sentiments

verb_sentiment_analysis = lambda doc:␣
↪word_level_sentiment_analysis(verb_search(doc))
adjective_sentiment_analysis = lambda doc:␣
↪word_level_sentiment_analysis(adjective_search(doc))

# Apply rule-based sentiment analysis to each document
sentiment_analysis = lambda doc: sia.polarity_scores(doc.text)
```

[6]:
```
# Add extensions to NLP pipeline
Doc.set_extension("noun_subjects", getter=noun_subject_search, force=True)
Doc.set_extension("robredo_subject", getter=robredo_subject_search, force=True)

Doc.set_extension("verbs", getter=verb_search, force=True)
Doc.set_extension("adjectives", getter=adjective_search, force=True)
```

```
Doc.set_extension("verb_sentiment", getter=verb_sentiment_analysis, force=True)
Doc.set_extension("adjective_sentiment", getter=adjective_sentiment_analysis,␣
 ↪force=True)
Doc.set_extension("sentiment", getter=sentiment_analysis, force=True)
```

[7]:
```
# Create a corpus and apply all processing steps
corpus = tweets[['id', 'handle', 'cleaned_text', 'supercleaned_text']]
corpus['document'] = list(nlp.pipe(corpus['cleaned_text']))
corpus
```

[7]:

	id	handle	cleaned_text	supercleaned_text
0	1461645389619027971	gonzoyyy	i will never apologize for being politically a…	never apologize politically active social medi…
1	1461645317158227969	charliekogure	kababuyan from them is the new pandemic if Len…	kababuyan new pandemic leni wins
2	1461645058688487429	beetookkk	Leni has been irrelevant since 2016.	leni irrelevant since 2016
3	1461644745134907396	adrianayalin	VP Leni thanked medical frontliners of Quezon …	vp leni thanked medical frontliners quezon con…
4	1461644676876828676	manigsaca	LENI is becoming like Pacquiao.. irrelevant	leni becoming like pacquiao irrelevant
…	…	…	…	…
1995	1461258152620400644	rapplerdotcom	LOOK: Presidential aspirant and VP Leni and he…	look presidential aspirant vp leni running mat…
1996	1461257854279512067	cnnphilippines	LOOK: Vice President Leni says she "had a comp…	look vice president leni says comprehensive fr…
1997	1461257600457052160	maracepeda	LOOK: Presidential aspirant and VP Leni and he…	look presidential aspirant vp leni running mat…
1998	1461257599471341569	myfuckinglegs	you guys saw vp leni having campaign (?) shoot…	guys saw vp leni campaign shoots tgis studio t…
1999	1461257355245457409	JustmeMajella	Please like, follow and share mga KakamPINK.Y…	

```
1999  please like follow share mga kakampinkyou requ…

                                                document
0     (i, will, never, apologize, for, being, politi…
1     (kababuyan, from, them, is, the, new, pandemic…
2         (Leni, has, been, irrelevant, since, 2016, .)
3     (VP, Leni, thanked, medical, frontliners, of, …
4     (LENI, is, becoming, like, Pacquiao, .., irrel…
…                                                   …
1995  (LOOK, :, Presidential, aspirant, and, VP, Len…
1996  (LOOK, :, Vice, President, Leni, says, she, ",…
1997  (LOOK, :, Presidential, aspirant, and, VP, Len…
1998  (you, guys, saw, vp, leni, having, campaign, (…
1999  (Please, like, ,, follow, and, share,  , mga, …

[2000 rows x 5 columns]
```

[8]:
```
# Retrieve all attributes from the documents and
# generate new columns from them
corpus['verbs'] = [d._.verbs for d in corpus['document']]
corpus['adjectives'] = [d._.adjectives for d in corpus['document']]
corpus['verb_sentiment'] = [d._.verb_sentiment for d in corpus['document']]
corpus['adjective_sentiment'] = [d._.adjective_sentiment for d in␣
↪corpus['document']]
corpus['verb_average_sentiment'] = [s['average_sentiment'] for s in␣
↪corpus['verb_sentiment']]
corpus['adjective_average_sentiment'] = [s['average_sentiment'] for s in␣
↪corpus['adjective_sentiment']]

corpus['sentiment'] = [d._.sentiment for d in corpus['document']]
corpus['neutral_sentiment'] = [s['neu'] for s in corpus['sentiment']]
corpus['positive_sentiment'] = [s['pos'] for s in corpus['sentiment']]
corpus['negative_sentiment'] = [s['neg'] for s in corpus['sentiment']]
corpus['compound_sentiment'] = [s['compound'] for s in corpus['sentiment']]
corpus.head()
```

[8]:
```
                   id         handle  \
0  1461645389619027971       gonzoyyy
1  1461645317158227969  charliekogure
2  1461645058688487429      beetookkk
3  1461644745134907396   adrianayalin
4  1461644676876828676      manigsaca

                                        cleaned_text  \
0  i will never apologize for being politically a…
1  kababuyan from them is the new pandemic if Len…
2               Leni has been irrelevant since 2016.
```

```
3  VP Leni thanked medical frontliners of Quezon …
4        LENI is becoming like Pacquiao.. irrelevant

                                  supercleaned_text  \
0  never apologize politically active social medi…
1                   kababuyan new pandemic leni wins
2                        leni irrelevant since 2016
3  vp leni thanked medical frontliners quezon con…
4             leni becoming like pacquiao irrelevant

                                             document  \
0  (i, will, never, apologize, for, being, politi…
1  (kababuyan, from, them, is, the, new, pandemic…
2      (Leni, has, been, irrelevant, since, 2016, .)
3  (VP, Leni, thanked, medical, frontliners, of, …
4  (LENI, is, becoming, like, Pacquiao, .., irrel…

                                               verbs  \
0  [apologize, apologize, tell, apologize, have, …
1                                             [win]
2                                               []
3                               [thank, speak, get]
4                                          [become]

                adjectives  \
0  [active, social, logical]
1                      [new]
2              [irrelevant]
3      [medical, continued]
4              [irrelevant]

                                      verb_sentiment  \
0  {'apologize': {'pos': 0.0, 'neg': 0.625, 'comp…
1  {'win': {'pos': 0.125, 'neg': 0.0, 'compound':…
2                           {'average_sentiment': 0}
3  {'thank': {'pos': 0.0, 'neg': 0.0, 'compound':…
4  {'become': {'pos': 0.0, 'neg': 0.0, 'compound'…

                                adjective_sentiment  verb_average_sentiment  \
0  {'active': {'pos': 0.375, 'neg': 0.0, 'compoun…               -0.078125
1  {'new': {'pos': 0.375, 'neg': 0.0, 'compound':…                0.062500
2  {'irrelevant': {'pos': 0.0, 'neg': 0.625, 'com…                0.000000
3  {'medical': {'pos': 0.125, 'neg': 0.0, 'compou…                0.000000
4  {'irrelevant': {'pos': 0.0, 'neg': 0.625, 'com…                0.000000

   adjective_average_sentiment  \
0                     0.166667
```

```
1                    0.187500
2                   -0.312500
3                    0.031250
4                   -0.312500

                                              sentiment  neutral_sentiment  \
0  {'neg': 0.089, 'neu': 0.782, 'pos': 0.129, 'co…              0.782
1  {'neg': 0.0, 'neu': 0.709, 'pos': 0.291, 'comp…              0.709
2  {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound…              1.000
3  {'neg': 0.0, 'neu': 0.776, 'pos': 0.224, 'comp…              0.776
4  {'neg': 0.0, 'neu': 0.667, 'pos': 0.333, 'comp…              0.667

   positive_sentiment  negative_sentiment  compound_sentiment
0               0.129               0.089              0.5736
1               0.291               0.000              0.5719
2               0.000               0.000              0.0000
3               0.224               0.000              0.8658
4               0.333               0.000              0.3612
```

[9]:
```
# Determine documents where Robredo is the main subject or otherwise
corpus['robredo_subject'] = [d._.robredo_subject for d in corpus['document']]
```

[10]:
```
# Remove documents where Leni is the main subject
corpus_robredo = corpus[corpus['robredo_subject'] == True].
 ↪reset_index(drop=True)
corpus_robredo.head()
```

[10]:
```
                   id         handle  \
0  1461645317158227969  charliekogure
1  1461645058688487429      beetookkk
2  1461644745134907396   adrianayalin
3  1461644676876828676      manigsaca
4  1461643691467046912    maharshalal

                                        cleaned_text  \
0  kababuyan from them is the new pandemic if Len…
1              Leni has been irrelevant since 2016.
2  VP Leni thanked medical frontliners of Quezon …
3       LENI is becoming like Pacquiao.. irrelevant
4  Leni's camp salivating on DDS differing opinio…

                                  supercleaned_text  \
0                  kababuyan new pandemic leni wins
1                        leni irrelevant since 2016
2  vp leni thanked medical frontliners quezon con…
3            leni becoming like pacquiao irrelevant
4  lenis camp salivating dds differing opinions c…
```

```
                                              document  \
0  (kababuyan, from, them, is, the, new, pandemic…
1      (Leni, has, been, irrelevant, since, 2016, .)
2  (VP, Leni, thanked, medical, frontliners, of, …
3  (LENI, is, becoming, like, Pacquiao, .., irrel…
4  (Leni, 's, camp, salivating, on, DDS, differin…

                                                verbs            adjectives  \
0                                               [win]                 [new]
1                                                  []           [irrelevant]
2                                [thank, speak, get]  [medical, continued]
3                                           [become]           [irrelevant]
4  [differ, find, vote, keep, salivate, go, happen]                 [less]

                                         verb_sentiment  \
0  {'win': {'pos': 0.125, 'neg': 0.0, 'compound':…
1                            {'average_sentiment': 0}
2  {'thank': {'pos': 0.0, 'neg': 0.0, 'compound':…
3  {'become': {'pos': 0.0, 'neg': 0.0, 'compound'…
4  {'differ': {'pos': 0.25, 'neg': 0.0, 'compound…

                                  adjective_sentiment  verb_average_sentiment  \
0  {'new': {'pos': 0.375, 'neg': 0.0, 'compound':…                0.062500
1  {'irrelevant': {'pos': 0.0, 'neg': 0.625, 'com…                0.000000
2  {'medical': {'pos': 0.125, 'neg': 0.0, 'compou…                0.000000
3  {'irrelevant': {'pos': 0.0, 'neg': 0.625, 'com…                0.000000
4  {'less': {'pos': 0.125, 'neg': 0.375, 'compoun…                0.071429

   adjective_average_sentiment  \
0                      0.18750
1                     -0.31250
2                      0.03125
3                     -0.31250
4                     -0.12500

                                              sentiment  neutral_sentiment  \
0  {'neg': 0.0, 'neu': 0.709, 'pos': 0.291, 'comp…              0.709
1  {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound…              1.000
2  {'neg': 0.0, 'neu': 0.776, 'pos': 0.224, 'comp…              0.776
3  {'neg': 0.0, 'neu': 0.667, 'pos': 0.333, 'comp…              0.667
4  {'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound…              1.000

   positive_sentiment  negative_sentiment  compound_sentiment  robredo_subject
0               0.291                 0.0              0.5719             True
1               0.000                 0.0              0.0000             True
2               0.224                 0.0              0.8658             True
```

```
3              0.333              0.0              0.3612           True
4              0.000              0.0              0.0000           True
```

[11]:
```
# Remove documents where Leni is not the main subject
corpus_non_robredo = corpus[corpus['robredo_subject'] == False].
↪reset_index(drop=True)
corpus_non_robredo.head()
```

[11]:
```
                   id           handle  \
0  1461645389619027971         gonzoyyy
1  1461644595129819138  jesuisteacherem
2  1461644288870137860     _aldrinperez
3  1461643840008327173  johnjoshcarillo
4  1461643462319566849       Einarr2013

                                        cleaned_text  \
0  i will never apologize for being politically a…
1  No disrespect to this guy. Though he claimed h…
2  Mga ganid. This clearly sets VP Leni apart fro…
3  dutertards &amp;amp; marcos apologists coming for …
4  No need to use intelligence to vote for Leni L…

                                     supercleaned_text  \
0  never apologize politically active social medi…
1  disrespect guy though claimed hes using pink s…
2  mga ganid clearly sets vp leni apart powerhung…
3  dutertards amp marcos apologists coming leni a…
4  need use intelligence vote leni lugaw voting o…

                                            document  \
0  (i, will, never, apologize, for, being, politi…
1  (No, disrespect, to, this, guy, ., Though, he,…
2  (Mga, ganid, ., This, clearly, sets, VP, Leni,…
3  (dutertards, &, amp, ;, marcos, apologists, co…
4  (No, need, to, use, intelligence, to, vote, fo…

                                              verbs  \
0  [apologize, apologize, tell, apologize, have, …
1                                  [claim, use]
2                                         [set]
3                [come, answer, event, formulate]
4                [use, vote, vote, mean, forfeit]

                adjectives  \
0  [active, social, logical]
1                    [bakit]
2    [hungry, incompetent]
```

```
3         [single, sensible]
4                         []

                                      verb_sentiment  \
0  {'apologize': {'pos': 0.0, 'neg': 0.625, 'comp…
1  {'claim': {'pos': 0.125, 'neg': 0.0, 'compound…
2  {'set': {'pos': 0.125, 'neg': 0.0, 'compound':…
3  {'come': {'pos': 0.0, 'neg': 0.0, 'compound': …
4  {'use': {'pos': 0.0, 'neg': 0.0, 'compound': 0…

                                adjective_sentiment  verb_average_sentiment  \
0  {'active': {'pos': 0.375, 'neg': 0.0, 'compoun…               -0.078125
1             {'bakit': {}, 'average_sentiment': 0}                0.031250
2  {'hungry': {'pos': 0.125, 'neg': 0.125, 'compo…                0.062500
3  {'single': {'pos': 0.0, 'neg': 0.0, 'compound'…               -0.015625
4                          {'average_sentiment': 0}                -0.015625

   adjective_average_sentiment  \
0                     0.166667
1                     0.000000
2                    -0.156250
3                     0.125000
4                     0.000000

                                       sentiment  neutral_sentiment  \
0  {'neg': 0.089, 'neu': 0.782, 'pos': 0.129, 'co…              0.782
1  {'neg': 0.0, 'neu': 0.9, 'pos': 0.1, 'compound…              0.900
2  {'neg': 0.17, 'neu': 0.658, 'pos': 0.171, 'com…              0.658
3  {'neg': 0.0, 'neu': 0.952, 'pos': 0.048, 'comp…              0.952
4  {'neg': 0.099, 'neu': 0.762, 'pos': 0.139, 'co…              0.762

   positive_sentiment  negative_sentiment  compound_sentiment  robredo_subject
0               0.129               0.089              0.5736            False
1               0.100               0.000              0.3252            False
2               0.171               0.170              0.0065            False
3               0.048               0.000              0.0772            False
4               0.139               0.099              0.2263            False
```

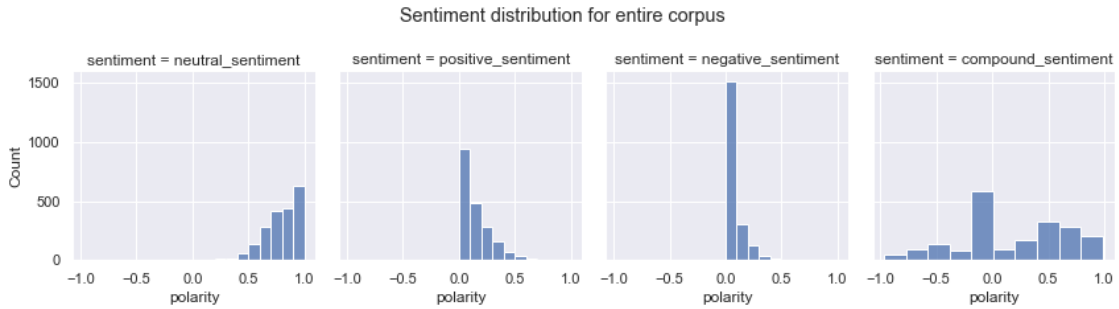
2 Data Analysis

[12]:
```
import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.style as style

style.use('fivethirtyeight')
sns.set(rc={'figure.figsize':(12,6)})
```

Sentiment Analysis

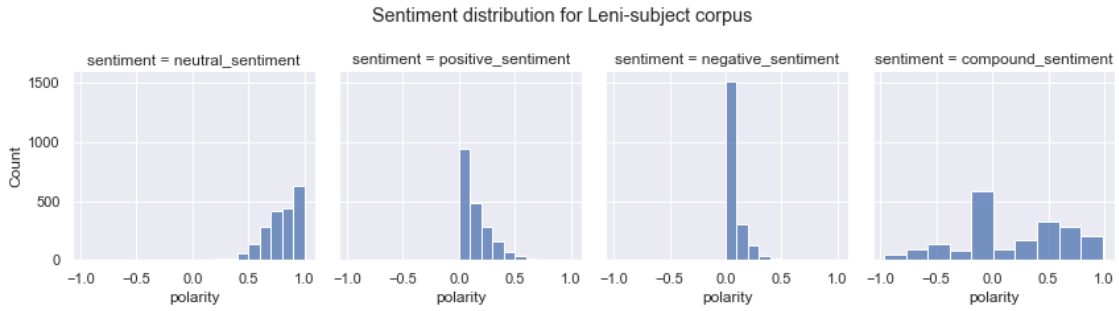
[13]:
```
# Unpivot sentiment columns
histogram_columns = ['id', 'neutral_sentiment', 'positive_sentiment',␣
↪'negative_sentiment', 'compound_sentiment']
corpus_sentiment = corpus[histogram_columns].melt('id', var_name='sentiment',␣
↪value_name='polarity')

# Visualize histogram plot of sentiment values for the entire corpus
corpus_sentiment_grid = sns.FacetGrid(corpus_sentiment, col="sentiment")
corpus_sentiment_grid.map(sns.histplot, "polarity", bins=10)
corpus_sentiment_grid.fig.suptitle('Sentiment distribution for entire corpus',␣
↪y=1.1)
```

[13]: Text(0.5, 1.1, 'Sentiment distribution for entire corpus')



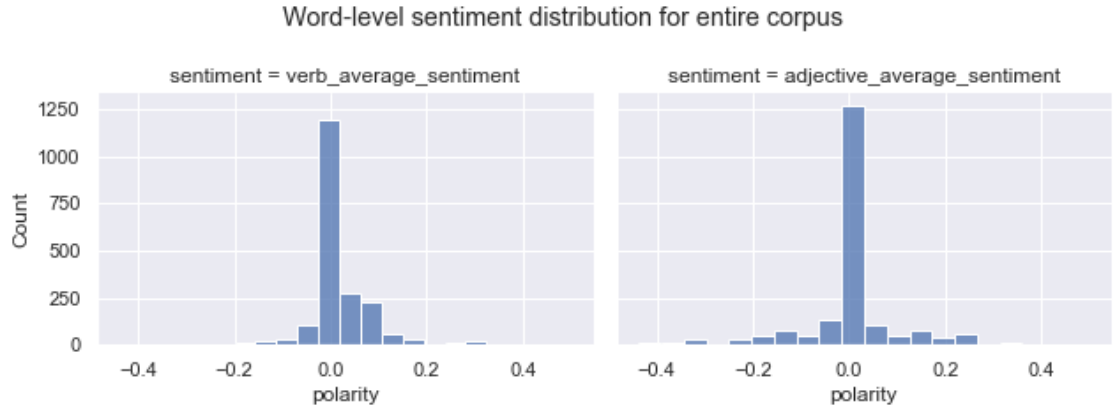
[14]:
```
# Unpivot sentiment columns
pos_histogram_columns = ['id', 'neutral_sentiment', 'positive_sentiment',␣
↪'negative_sentiment', 'compound_sentiment']
corpus_robredo_sentiment = corpus[histogram_columns].melt('id',␣
↪var_name='sentiment', value_name='polarity')

# Visualize histogram plot of sentiment values for the Leni-subject corpus
corpus_robredo_sentiment_grid = sns.FacetGrid(corpus_robredo_sentiment,␣
↪col="sentiment")
corpus_robredo_sentiment_grid.map(sns.histplot, "polarity", bins=10)
corpus_robredo_sentiment_grid.fig.suptitle('Sentiment distribution for␣
↪Leni-subject corpus', y=1.1)
```

[14]: Text(0.5, 1.1, 'Sentiment distribution for Leni-subject corpus')



[15]:
```
# Unpivot sentiment columns
pos_histogram_columns = ['id', 'verb_average_sentiment',␣
↪'adjective_average_sentiment']
corpus_pos_sentiment = corpus[pos_histogram_columns].melt('id',␣
↪var_name='sentiment', value_name='polarity')

# Visualize histogram plot of sentiment values for the entire corpus
corpus_pos_sentiment_grid = sns.FacetGrid(corpus_pos_sentiment,␣
↪col="sentiment", aspect=1.5)
corpus_pos_sentiment_grid.map(sns.histplot, "polarity", bins=20)
corpus_pos_sentiment_grid.fig.suptitle('Word-level sentiment distribution for␣
↪entire corpus', y=1.1)
```

[15]:
```
Text(0.5, 1.1, 'Word-level sentiment distribution for entire corpus')
```



Term frequency analysis

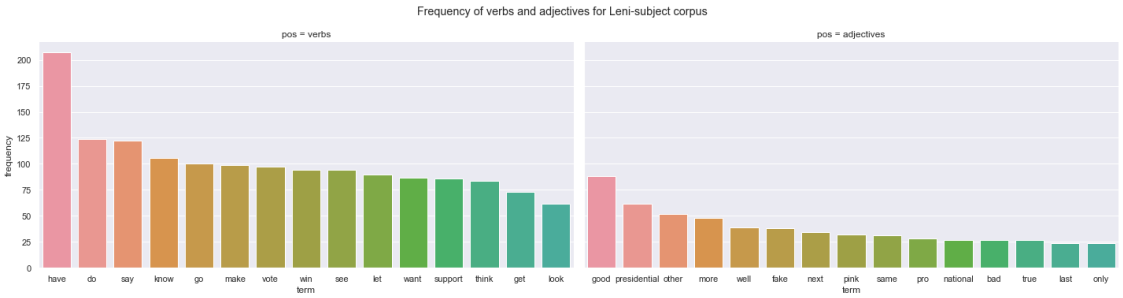
[16]:
```
from sklearn.feature_extraction.text import CountVectorizer
```

[17]:
```
# Create a token frequency table
def tokens_to_frequency(tokens, pos=''):
    vectorizer = CountVectorizer(analyzer=lambda x: x)
    dtm = vectorizer.fit_transform(tokens)
    total = dtm.sum(axis=0)
    frequency = [(w, total[0, i]) for w, i in vectorizer.vocabulary_.items()]
    frequency = pd.DataFrame(frequency, columns=['term', 'frequency'])
    frequency = frequency.sort_values(by='frequency', ascending=False).
 ↪reset_index(drop=True)
    frequency['pos'] = pos
    return frequency
```

[18]:
```
# Unpivot frequency columns
def generate_pos_token_frequency(corpus, cols):
    t2fs = []
    for col in cols:
        t2fs.append(tokens_to_frequency(corpus[col], pos=col))
    df_frequency = pd.concat(t2fs)
    df_frequency = df_frequency.groupby('pos').head(15).reset_index(drop=True)
    return df_frequency
```

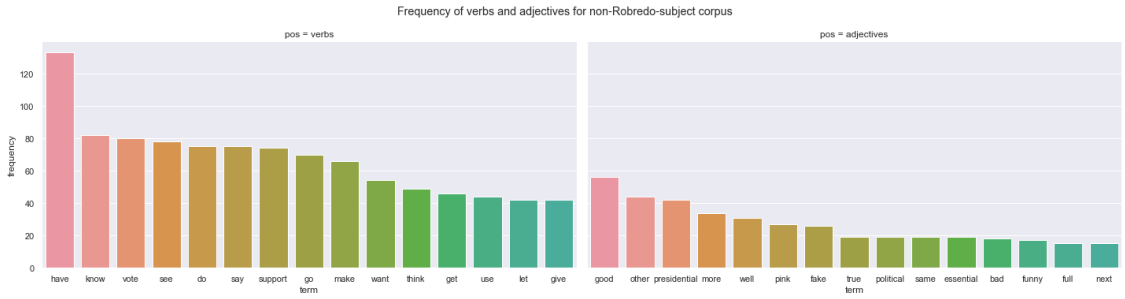
[19]:
```
# Create frequency table
corpus_robredo_frequency = generate_pos_token_frequency(corpus, ['verbs',␣
 ↪'adjectives'])

# Visualize histogram plot of frequency values for the Leni corpus
corpus_robredo_frequency_plot = sns.catplot(
    data=corpus_robredo_frequency,
    x="term", y="frequency", col="pos",
    kind="bar", sharex=False, aspect=2,
)
corpus_robredo_frequency_plot.fig.suptitle('Frequency of verbs and adjectives␣
 ↪for Leni-subject corpus', y=1.05)
```

[19]:
```
Text(0.5, 1.05, 'Frequency of verbs and adjectives for Leni-subject corpus')
```



[20]:
```
# Create frequency table
corpus_non_robredo_frequency = generate_pos_token_frequency(corpus_non_robredo,␣
↪['verbs', 'adjectives'])

# Visualize histogram plot of frequency values for the Leni corpus
corpus_non_robredo_frequency_plot = sns.catplot(
    data=corpus_non_robredo_frequency,
    x="term", y="frequency", col="pos",
    kind="bar", sharex=False, aspect=2,
)
corpus_non_robredo_frequency_plot.fig.suptitle('Frequency of verbs and␣
↪adjectives for non-Robredo-subject corpus', y=1.05)
```

[20]:
```
Text(0.5, 1.05, 'Frequency of verbs and adjectives for non-Robredo-subject
corpus')
```



Topic Modelling

[21]:
```
!pip install pyLDAvis
from gensim.corpora.dictionary import Dictionary
from gensim.models.ldamodel import LdaModel
from gensim.models import CoherenceModel
import pyLDAvis, pyLDAvis.gensim_models
```

```
Requirement already satisfied: pyLDAvis in d:\anaconda3\lib\site-packages
(3.3.1)
Requirement already satisfied: gensim in d:\anaconda3\lib\site-packages (from
pyLDAvis) (4.1.2)
Requirement already satisfied: scikit-learn in d:\anaconda3\lib\site-packages
(from pyLDAvis) (0.24.2)
Requirement already satisfied: joblib in d:\anaconda3\lib\site-packages (from
pyLDAvis) (1.1.0)
Requirement already satisfied: sklearn in d:\anaconda3\lib\site-packages (from
pyLDAvis) (0.0)
Requirement already satisfied: jinja2 in d:\anaconda3\lib\site-packages (from
pyLDAvis) (2.11.3)
```

```
Requirement already satisfied: scipy in d:\anaconda3\lib\site-packages (from
pyLDAvis) (1.7.1)
Requirement already satisfied: funcy in d:\anaconda3\lib\site-packages (from
pyLDAvis) (1.17)
Requirement already satisfied: numpy>=1.20.0 in d:\anaconda3\lib\site-packages
(from pyLDAvis) (1.20.3)
Requirement already satisfied: future in d:\anaconda3\lib\site-packages (from
pyLDAvis) (0.18.2)
Requirement already satisfied: numexpr in d:\anaconda3\lib\site-packages (from
pyLDAvis) (2.7.3)
Requirement already satisfied: setuptools in d:\anaconda3\lib\site-packages
(from pyLDAvis) (58.0.4)
Requirement already satisfied: pandas>=1.2.0 in d:\anaconda3\lib\site-packages
(from pyLDAvis) (1.3.4)
Requirement already satisfied: pytz>=2017.3 in d:\anaconda3\lib\site-packages
(from pandas>=1.2.0->pyLDAvis) (2021.3)
Requirement already satisfied: python-dateutil>=2.7.3 in d:\anaconda3\lib\site-
packages (from pandas>=1.2.0->pyLDAvis) (2.8.2)
Requirement already satisfied: six>=1.5 in d:\anaconda3\lib\site-packages (from
python-dateutil>=2.7.3->pandas>=1.2.0->pyLDAvis) (1.16.0)
Requirement already satisfied: smart-open>=1.8.1 in d:\anaconda3\lib\site-
packages (from gensim->pyLDAvis) (5.2.1)
Requirement already satisfied: Cython==0.29.23 in d:\anaconda3\lib\site-packages
(from gensim->pyLDAvis) (0.29.23)
Requirement already satisfied: MarkupSafe>=0.23 in d:\anaconda3\lib\site-
packages (from jinja2->pyLDAvis) (1.1.1)
Requirement already satisfied: threadpoolctl>=2.0.0 in d:\anaconda3\lib\site-
packages (from scikit-learn->pyLDAvis) (2.2.0)
```

[22]:
```
corpus['supercleaned_text']
```

[22]:
```
0       never apologize politically active social medi…
1                         kababuyan new pandemic leni wins
2                               leni irrelevant since 2016
3       vp leni thanked medical frontliners quezon con…
4                  leni becoming like pacquiao irrelevant
                               …
1995    look presidential aspirant vp leni running mat…
1996    look vice president leni says comprehensive fr…
1997    look presidential aspirant vp leni running mat…
1998    guys saw vp leni campaign shoots tgis studio t…
1999    please like follow share mga kakampinkyou requ…
Name: supercleaned_text, Length: 2000, dtype: object
```

[23]:
```
# Create a corpus from the documents
split_texts = [[token.text for token in doc] for doc in corpus['document']]
dictionary = Dictionary(split_texts)
```

```
lda_corpus = [dictionary.doc2bow(text) for text in split_texts]
```

[24]:
```
# Create an LDA model with 4 topics from the generated corpus
lda_model = LdaModel(
    corpus=lda_corpus,
    id2word=dictionary,
    num_topics=4,
    random_state=110
)
```

[25]:
```
# Visualize the topic models and figure out the themes
pyLDAvis.enable_notebook()
vis = pyLDAvis.gensim_models.prepare(lda_model, lda_corpus, dictionary)
vis
```

[25]:
```
PreparedData(topic_coordinates=              x         y  topics  cluster
Freq
topic
0      0.007718  0.003702       1        1  34.120719
1      0.031815  0.021418       2        1  31.383360
3      0.002641 -0.039481       3        1  20.305506
2     -0.042175  0.014362       4        1  14.190415, topic_info=    Term
Freq        Total Category  logprob  loglift
241    !   402.000000   402.000000  Default  30.0000  30.0000
0      .  2060.000000  2060.000000  Default  29.0000  29.0000
81     ,  1278.000000  1278.000000  Default  28.0000  28.0000
299    "   219.000000   219.000000  Default  27.0000  27.0000
1      a   557.000000   557.000000  Default  26.0000  26.0000
..   …          …           …      …       …        …
178  not    28.217809   218.244930   Topic4  -5.5865  -0.0931
265   as    24.633455   156.175029   Topic4  -5.7223   0.1057
84     ?    29.438460   356.732824   Topic4  -5.5441  -0.5421
8    for    31.339791   558.647076   Topic4  -5.4815  -0.9280
17    on    25.784017   224.460380   Topic4  -5.6767  -0.2113

[330 rows x 6 columns], token_table=      Topic      Freq        Term
term
207       1  0.493825
207       2  0.291969
207       3  0.176623
207       4  0.036046
4948      2  0.736975
…     …       …          …
480       4  0.118680          ’s
3453      1  0.173591           ₱
3453      3  0.694362           ₱
3453      4  0.173591           ₱
```

```
5649      2  0.737209

[531 rows x 3 columns], R=30, lambda_step=0.01, plot_opts={'xlab': 'PC1',
'ylab': 'PC2'}, topic_order=[1, 2, 4, 3])
```

[ ]: