Sentiment Analysis-Part 1

Code

Jeff Gross

Install packages

Hide

```
install.packages("readr")
install.packages("magrittr")
install.packages("qdap")
install.packages("NLP")
install.packages("tm")
install.packages("metricsgraphics")
install.packages("tidyr")
install.packages("tidytext")
#install.packages("plyr")
install.packages("ggthemes")
install.packages("dplyr")
install.packages("ggplot")
install.packages("wordcloud")
install.packages("RColorBrewer")
install.packages("radarchart")
install.packages("treemap")

library(readr)
library(magrittr)
library(qdap)
library(NLP)
library(tm)
library(metricsgraphics)
library(tidyr)
library(tidytext)
#library(plyr)
library(ggthemes)
library(dplyr)
library(ggplot2)
library(wordcloud)
library(RColorBrewer)
library(radarchart)
library(treemap)
```

Task: Visualize the polarity of a conversation.

Hide

```
library(readr)
library(magrittr)
library(qdap)
system("java -version")
```

```
java version "1.8.0_151"
Java(TM) SE Runtime Environment (build 1.8.0_151-b12)
Java HotSpot(TM) 64-Bit Server VM (build 25.151-b12, mixed mode)
```

Hide

```
Sys.setenv(JAVA_HOME = "C:/Program Files/Java/jre1.8.0_151/")
text_df <- read_csv("C:/Users/Y/Documents/1.R_scripts/ML/NLP/Sentiment_Analysis/text_df.csv")
```

```
Parsed with column specification:
cols(
  person = col_character(),
  text = col_character()
)
```

Hide

```
# Examine the text data
text_df
```

person <chr>	text <chr>
Nick	DataCamp courses are the best
Jonathan	I like talking to students
Martijn	Other online data science curricula are boring.
Nicole	What is for lunch?
Nick	DataCamp has lots of great content!
Jonathan	Students are passionate and are excited to learn
Martijn	Other data science curriculum is hard to learn and difficult to understand
Nicole	I think the food here is good.

8 rows

Hide

```
# Calc overall polarity score
text_df %$% polarity(text)
```

```
  all total.sentences total.words ave.polarity sd.polarity stan.mean.polarity
1 all               8          54        0.179       0.452              0.396
```

Hide

```
# Calc polarity score by person
(datacamp_conversation <- text_df %$% polarity(text, person))
```

```
    person total.sentences total.words ave.polarity sd.polarity stan.mean.polarity
1 Jonathan               2          13        0.577       0.184             3.141
2  Martijn               2          19       -0.478       0.141            -3.388
3     Nick               2          11        0.428       0.028            15.524
4   Nicole               2          11        0.189       0.267             0.707
```

Hide

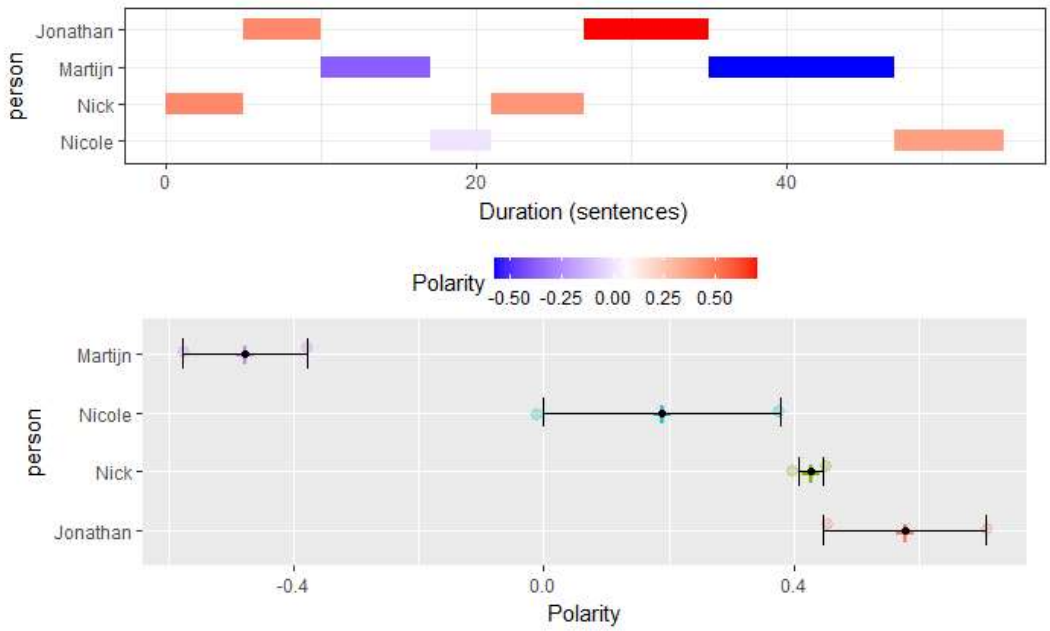
```
# Counts table from datacamp_conversation
counts(datacamp_conversation)
```

```
    person wc polarity           pos.words       neg.words
text.var
1     Nick  5    0.447                best               -                                        DataCamp courses are
the best
2 Jonathan  5    0.447                like               -                                           I like talking to
students
3  Martijn  7   -0.378                   -          boring                       Other online data science curricula ar
e boring.
4   Nicole  4    0.000                   -               -                                                   What is f
or lunch?
5     Nick  6    0.408               great               -                                  DataCamp has lots of great
content!
6 Jonathan  8    0.707 passionate, excited               -                         Students are passionate and are excited
to learn
7  Martijn 12   -0.577                   - hard, difficult Other data science curriculum is hard to learn and difficult to u
nderstand
8   Nicole  7    0.378                good               -                                       I think the food here
is good.
```

Hide

```
# Plot the conversation polarity
plot(datacamp_conversation)
```

```
`show_guide` has been deprecated. Please use `show.legend` instead.`show_guide` has been deprecated. Please use `show.legend
` instead.
```



Task: Construct a DTM after cleaning the corpus.

Hide

```
library(NLP)
library(tm)
library(qdap)
tm_define <- c("Text mining is the process of distilling actionable insights from text.",                    "Sentiment a
nalysis represents the set of tools to extract an author's feelings towards a subject.")
# clean_corpus(),
clean_corpus_2 <- function(corpus){
  corpus <- tm_map(corpus, content_transformer(replace_abbreviation))
  corpus <- tm_map(corpus, removePunctuation)
  corpus <- tm_map(corpus, removeNumbers)
  corpus <- tm_map(corpus, removeWords, c(stopwords("en"), "coffee"))
  corpus <- tm_map(corpus, content_transformer(tolower))
  corpus <- tm_map(corpus, stripWhitespace)
  return(corpus)
}
tm_define <- c("Text mining is the process of distilling actionable insights from text.", "Sentiment analysis represents the
 set of tools to extract an author's feelings towards a subject.")
# Create a VectorSource
tm_vector <- VectorSource(tm_define)
tm_vector
```

```
$encoding
[1] ""

$length
[1] 2

$position
[1] 0

$reader
function (elem, language, id)
{
    if (!is.null(elem$uri))
        id <- basename(elem$uri)
    PlainTextDocument(elem$content, id = id, language = language)
}
<environment: namespace:tm>

$content
[1] "Text mining is the process of distilling actionable insights from text."
[2] "Sentiment analysis represents the set of tools to extract an author's feelings towards a subject."

attr(,"class")
[1] "VectorSource" "SimpleSource" "Source"
```

Hide

```
# Apply VCorpus
tm_corpus <- VCorpus(tm_vector)
tm_corpus
```

```
<<VCorpus>>
Metadata:  corpus specific: 0, document level (indexed): 0
Content:  documents: 2
```

Hide

```
# Examine the first document's contents
content(tm_corpus[[1]])
```

```
[1] "Text mining is the process of distilling actionable insights from text."
```

Hide

```
# Clean the text
tm_clean <- clean_corpus_2(tm_corpus)
# Reexamine the contents of the first doc
content(tm_clean[[1]])
```

```
[1] "text mining process distilling actionable insights text"
```

Task: Clean and create DTM for coffee tweets.

Hide

```
library(readr)
coffee <- read_csv("C:/Users/Y/Documents/1.R_scripts/ML/NLP/Sentiment_Analysis/coffee.csv", col_types = cols(created = col_n
umber(),
    id = col_number(), replyToSID = col_number()))
head(coffee)
```

doc_id <int>
1
2

doc_id <int>
3
4
5
6

6 rows | 1-1 of 15 columns

Hide

```
dim(coffee)
```

```
[1] 1000   15
```

Hide

```
# clean_corpus()
clean_corpus <- function(corpus){
  corpus <- tm_map(corpus, content_transformer(replace_abbreviation))
  corpus <- tm_map(corpus, removePunctuation)
  corpus <- tm_map(corpus, removeNumbers)
  corpus <- tm_map(corpus, removeWords, c(stopwords("en"), "coffee"))
  corpus <- tm_map(corpus, content_transformer(tolower))
  corpus <- tm_map(corpus, stripWhitespace)
  return(corpus)
}
# Isolate text from tweets: coffee_tweets
#coffee_tweets <- tweets$text
# Create a DataframeSource on columns 2 and 3: df_source
coffee_source <- DataframeSource(coffee[,1:2])
head(coffee_source)
```

doc_id <int>
1
2
3
4
5
6

6 rows | 1-1 of 2 columns

Hide

```
# Convert df_source to a corpus: df_corpus
coffee_corpus <- VCorpus(coffee_source)
clean_text <- clean_corpus(coffee_corpus)
clean_text
```

```
<<VCorpus>>
Metadata:  corpus specific: 0, document level (indexed): 0
Content:  documents: 1000
```

Hide

```
# clean_text is pre-defined
clean_text
```

```
<<VCorpus>>
Metadata:  corpus specific: 0, document level (indexed): 0
Content:  documents: 1000
```

Hide

```
# Create tf_dtm
tf_dtm <- DocumentTermMatrix(clean_text)
# Create tf_dtm_m
tf_dtm_m <- as.matrix(tf_dtm)
# Dimensions of DTM matrix
dim(tf_dtm_m)
```

```
[1] 1000 3098
```

Hide

```
# Subset part of tf_dtm_m for comparison
tf_dtm_m[16:20, 2975:2985]
```

```
    Terms
Docs went were west westin westside wet wfriends what whatever whatislifeee whats
  16    0    0    0      0        0   0        0    0        0            0     0
  17    0    0    0      0        0   0        0    0        0            0     0
  18    0    0    0      0        0   0        0    0        0            0     0
  19    0    0    0      0        0   0        0    0        0            0     0
  20    0    0    0      0        0   0        0    0        0            0     0
```

Task : Construct a visual from 3 million tweets mentioning “#sb”.

Hide

```
library(magrittr)
library(metricsgraphics)
sb_words <- read_csv("C:/Users/Y/Documents/1.R_scripts/ML/NLP/Sentiment_Analysis/sb_words.csv")
```

```
Parsed with column specification:
cols(
  word = col_character(),
  freq = col_integer(),
  rank = col_integer()
)
```

Hide

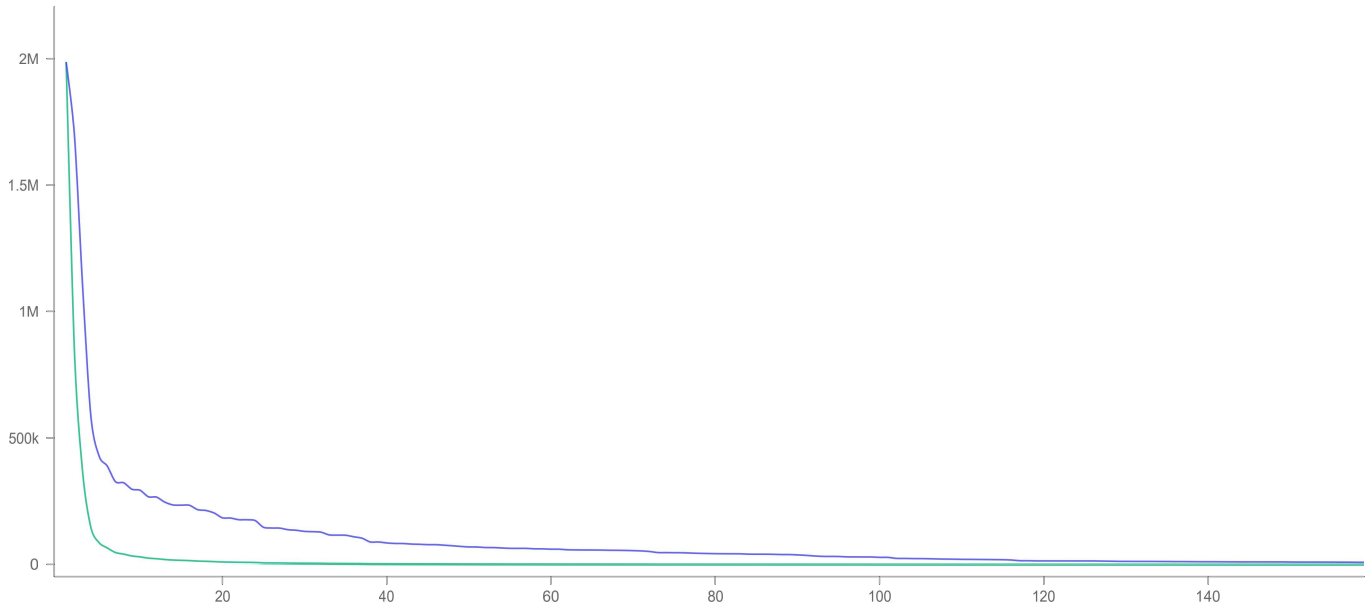
```
# Examine sb_words
head(sb_words)
```

word <chr>	freq <int>	rank <int>
sb	1984423	1
rt	1700564	2
the	1101899	3
to	588803	4
a	428598	5
for	388390	6

6 rows

Hide

```
# Create expectations
sb_words$expectations <- sb_words %$%
  {freq / rank}
# Create metrics plot
sb_plot <- mjs_plot(sb_words, x = rank, y = freq, show_rollover_text = FALSE)
# Add 1st line
sb_plot <- mjs_line(sb_plot)
# Add 2nd line
sb_plot <- mjs_add_line(sb_plot, expectations)
# Add legend
sb_plot <- mjs_add_legend(sb_plot, legend = c("Frequency", "Expectation"))
# Display plot
sb_plot
```



— Frequency — Expectation

Task: Perform a simple polarity calculation.

Hide

```
# Example statements
positive <- "DataCamp courses are good for learning"
# Calculate polarity of both statements
(pos_score <-polarity(positive))
```

```
  all total.sentences total.words ave.polarity sd.polarity stan.mean.polarity
1 all               1           6        0.408          NA                NA
```

Hide

```
# Get counts
(pos_counts <- counts(pos_score))
```

```
  all wc polarity pos.words neg.words                               text.var
1 all  6    0.408      good         - DataCamp courses are good for learning
```

Hide

```
# Number of positive words
n_good <- length(pos_counts$pos.words[[1]])

# Total number of words
n_words <- pos_counts$wc

# Verify polarity score
n_good / sqrt(n_words)
```

```
[1] 0.4082483
```

Task: Use valence shifters, amplifiers and negating words, which tell you about the author's emotional intent to create a plot.

Result: 'Extremely good' is more positive than 'very good', which is more positive than 'good', which is more positive than 'quite good'.

Hide

```
library(readr)
library(qdap)
library(magrittr)
conversation <- read_csv("C:/Users/Y/Documents/1.R_scripts/ML/NLP/Sentiment_Analysis/conversation.csv")
```

```
Parsed with column specification:
cols(
  student = col_character(),
  text = col_character()
)
```

Hide

```
# Examine conversation
conversation
```

student	text
<chr>	<chr>
Martijn	This restaurant is never bad
Nick	The lunch was very good
Nicole	It was awful I got food poisoning and was extremely ill

3 rows

Hide

```
# Polarity - All
polarity(conversation$text)
```

```
  all total.sentences total.words ave.polarity sd.polarity stan.mean.polarity
1 all               3          21        0.317       0.565              0.561
```

Hide

```
# Polarity - Grouped
student_pol <- conversation %$%
  polarity(text, student)
# Student results
scores(student_pol)
```

```
  student total.sentences total.words ave.polarity sd.polarity stan.mean.polarity
1 Martijn               1           5        0.447          NA                 NA
2    Nick               1           5        0.805          NA                 NA
3  Nicole               1          11       -0.302          NA                 NA
```

Hide

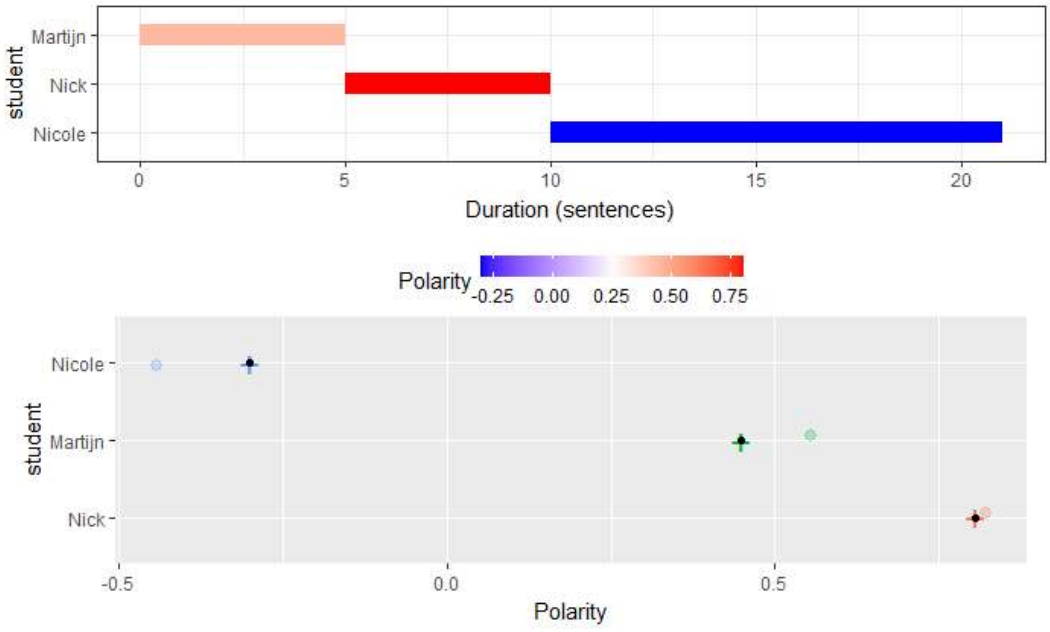
```
# Sentence by sentence
counts(student_pol)
```

```
  student wc polarity pos.words neg.words                                                text.var
1 Martijn  5    0.447         -       bad                            This restaurant is never bad
2    Nick  5    0.805      good         -                                 The lunch was very good
3  Nicole 11   -0.302         -     awful It was awful I got food poisoning and was extremely ill
```

Hide

```
# qdap plot
plot(student_pol)
```

```
`show_guide` has been deprecated. Please use `show.legend` instead.`show_guide` has been deprecated. Please use `show.legend
` instead.
```



Task: Examine the existing word data frame objects

Hide

```
text <- read_csv("C:/Users/Y/Documents/1.R_scripts/ML/NLP/Sentiment_Analysis/text.csv")
```

```
Parsed with column specification:
cols(
  speaker = col_character(),
  words = col_character()
)
```

Hide

```
# Examine the key.pol
key.pol
```

x	y
<chr>	<dbl>

	x <chr>	y <dbl>
1	a plus	1
2	abnormal	-1
3	abolish	-1
4	abominable	-1
5	abominably	-1
6	abominate	-1
7	abomination	-1
8	abort	-1
9	aborted	-1
10	aborts	-1

1-10 of 6,779 rows Previous **1** 2 3 4 5 6 … 100 Next

Hide

```
# Negators
negation.words
```

```
 [1] "ain't"     "aren't"    "can't"     "couldn't"  "didn't"    "doesn't"   "don't"     "hasn't"    "isn't"
[10] "mightn't"  "mustn't"   "neither"   "never"     "no"        "nobody"    "nor"       "not"       "shan't"
[19] "shouldn't" "wasn't"    "weren't"   "won't"     "wouldn't"
```

Hide

```
# Amplifiers
amplification.words
```

```
 [1] "acute"         "acutely"       "certain"       "certainly"     "colossal"      "colossally"
 [7] "deep"          "deeply"        "definite"      "definitely"    "enormous"      "enormously"
[13] "extreme"       "extremely"     "great"         "greatly"       "heavily"       "heavy"
[19] "high"          "highly"        "huge"          "hugely"        "immense"       "immensely"
[25] "incalculable"  "incalculably"  "massive"       "massively"     "more"          "particular"
[31] "particularly"  "purpose"       "purposely"     "quite"         "real"          "really"
[37] "serious"       "seriously"     "severe"        "severely"      "significant"   "significantly"
[43] "sure"          "surely"        "true"          "truly"         "vast"          "vastly"
[49] "very"
```

Hide

```
# De-amplifiers
deamplification.words
```

```
 [1] "barely"       "faintly"      "few"          "hardly"       "little"       "only"         "rarely"
 [8] "seldom"       "slightly"     "sparesly"     "sporadically" "very few"     "very little"
```

Hide

```
# Examine
text
```

speaker <chr>	words <chr>
beyonce	I know I dont understand Just how your love can do what no one else can
jay_z	They cant figure him out they like hey, is he insane

2 rows

Hide

```
# Explicit polarity parameters
polarity(
  text.var       = text$words,
  grouping.var   = text$speaker,
  polarity.frame = key.pol,
  negators       = negation.words,
  amplifiers     = amplification.words,
  deamplifiers   = deamplification.words
)
```

```
  speaker total.sentences total.words ave.polarity sd.polarity stan.mean.polarity
1 beyonce               1          16         0.25          NA                 NA
2   jay_z               1          11         0.00          NA                 NA
```

Task: Take the specific features of the text you're analyzing into account.

Result: It's important to take the specific features of the text you're analyzing into account so that you can make sure your results are accurate.

Hide

```
stressed_out <- c("I wish I found some better sounds no ones ever heard\nI wish I had a better voice that sang some better w
ords\nI wish I found some chords in an order that is new\nI wish I didnt have to rhyme every time I sang\nI was told when I
 get older all my fears would shrink\nBut now Im insecure and I care what people think\nMy names Blurryface and I care what
 you think\nMy names Blurryface and I care what you think\nWish we could turn back time, to the good old days\nWhen our momm
a sang us to sleep but now were stressed out\nWish we could turn back time to the good old days\nWhen our momma sang us to s
leep but now were stressed out\nWere stressed out\nSometimes a certain smell will take me back to when I was young\nHow come
 Im never able to identify where its coming from\nId make a candle out of it if I ever found it\nTry to sell it never sell o
ut of it Id probably only sell one\nItd be to my brother, cause we have the same nose\nSame clothes homegrown a stones throw
 from a creek we used to roam\nBut it would remind us of when nothing really mattered\nOut of student loans and tree-house h
omes we all would take the latter\nMy names Blurryface and I care what you think\nMy names Blurryface and I care what you th
ink\nWish we could turn back time, to the good old days\nWhen our momma sang us to sleep but now were stressed out\nWish we
 could turn back time, to the good old days\nWhen our momma sang us to sleep but now were stressed out\nWe used to play pret
end, give each other different names\nWe would build a rocket ship and then wed fly it far away\nUsed to dream of outer spac
e but now theyre laughing at our face #\nSaying, Wake up you need to make money\nYeah\nWe used to play pretend give each oth
er different names\nWe would build a rocket ship and then wed fly it far away\nUsed to dream of outer space but now theyre l
aughing at our face\nSaying, Wake up, you need to make money\nYeah\nWish we could turn back time, to the good old days\nWhen
 our momma sang us to sleep but now were stressed out\nWish we could turn back time, to the good old days\nWhen our momma sa
ng us to sleep but now were stressed out\nUsed to play pretend, used to play pretend bunny\nWe used to play pretend wake up,
 you need the money\nUsed to play pretend used to play pretend bunny\nWe used to play pretend, wake up, you need the money\n
We used to play pretend give each other different names\nWe would build a rocket ship and then wed fly it far away\nUsed to
 dream of outer space but now theyre laughing at our face\nSaying, Wake up, you need to make money\nYeah")
# stressed_out has been pre-defined
head(stressed_out)
```

```
[1] "I wish I found some better sounds no ones ever heard\nI wish I had a better voice that sang some better words\nI wish I
found some chords in an order that is new\nI wish I didnt have to rhyme every time I sang\nI was told when I get older all m
y fears would shrink\nBut now Im insecure and I care what people think\nMy names Blurryface and I care what you think\nMy na
mes Blurryface and I care what you think\nWish we could turn back time, to the good old days\nWhen our momma sang us to slee
p but now were stressed out\nWish we could turn back time to the good old days\nWhen our momma sang us to sleep but now were
stressed out\nWere stressed out\nSometimes a certain smell will take me back to when I was young\nHow come Im never able to
identify where its coming from\nId make a candle out of it if I ever found it\nTry to sell it never sell out of it Id probab
ly only sell one\nItd be to my brother, cause we have the same nose\nSame clothes homegrown a stones throw from a creek we u
sed to roam\nBut it would remind us of when nothing really mattered\nOut of student loans and tree-house homes we all would
take the latter\nMy names Blurryface and I care what you think\nMy names Blurryface and I care what you think\nWish we could
turn back time, to the good old days\nWhen our momma sang us to sleep but now were stressed out\nWish we could turn back tim
e, to the good old days\nWhen our momma sang us to sleep but now were stressed out\nWe used to play pretend, give each other
different names\nWe would build a rocket ship and then wed fly it far away\nUsed to dream of outer space but now theyre laug
hing at our face #\nSaying, Wake up you need to make money\nYeah\nWe used to play pretend give each other different names\nW
e would build a rocket ship and then wed fly it far away\nUsed to dream of outer space but now theyre laughing at our face\n
Saying, Wake up, you need to make money\nYeah\nWish we could turn back time, to the good old days\nWhen our momma sang us to
sleep but now were stressed out\nWish we could turn back time, to the good old days\nWhen our momma sang us to sleep but now
were stressed out\nUsed to play pretend, used to play pretend bunny\nWe used to play pretend wake up, you need the money\nUs
ed to play pretend used to play pretend bunny\nWe used to play pretend, wake up, you need the money\nWe used to play pretend
give each other different names\nWe would build a rocket ship and then wed fly it far away\nUsed to dream of outer space but
now theyre laughing at our face\nSaying, Wake up, you need to make money\nYeah"
```

Hide

```
# Basic lexicon score
polarity(stressed_out)
```

```
  all total.sentences total.words ave.polarity sd.polarity stan.mean.polarity
1 all               1         518       -0.255          NA                NA
```

Hide

```
# Check the subjectivity lexicon
key.pol[grep("stress", x)]
```

x <chr>	y <dbl>
distress	-1
distressed	-1
distressing	-1
distressingly	-1
mistress	-1
stress	-1
stresses	-1
stressful	-1
stressfully	-1

9 rows

Hide

```
# New lexicon
custom_pol <- sentiment_frame(positive.words, c(negative.words, "stressed", "turn back"))
# Compare new score
polarity(stressed_out, polarity.frame = custom_pol)
```

```
  all total.sentences total.words ave.polarity sd.polarity stan.mean.polarity
1 all               1         518       -0.826          NA                NA
```