Lexical Sentiment Analysis

Wouter van Atteveldt

June 3, 2016

Dictionaries of positive and negative terms can be used to do sentiment analysis, assuming that a document with many positive terms will have a more positive sentiment.

Lexicon and data

For this exercise, we will use the Pittsburgh sentiment dictionary and the Amazon automotive reviews as described in the 'Getting Sentiment Resources' hand-out. These files can be directly downloaded from http://rawgit.com/vanatteveldt/learningr/master/data/reviews.rds (reviews) and http://rawgit.com/vanatteveldt/learningr/master/data/lexicon.rds (lexicon).

```
reviews = readRDS("data/reviews.rds")
lexicon = readRDS("data/lexicon.rds")
```

Applying Sentiment Dictionary to DTM

You can directly apply a dictionary to the document term matrix by summing the columns that match each category.

First, we create the document term matrix consisting of all reviews:

```
library(RTextTools)
dtm = create_matrix(reviews[c("summary", "reviewText")], language="english", stemWords=T)
```

And we select the words that are in the negative or positive category:

```
pos_words = lexicon$word1[lexicon$priorpolarity == "positive"]
neg_words = lexicon$word1[lexicon$priorpolarity == "negative"]
```

Now, we use these words to subset the dtm, and use `row_sums` to sum all words in the same category for each document:

```
library(slam)
reviews$npos = row_sums(dtm[, colnames(dtm) %in% pos_words])
reviews$nneg = row_sums(dtm[, colnames(dtm) %in% neg_words])
```

Finally, we can calculate a sentiment score, for example the number of positive minus negative words normalized by the total number of sentiment words:

```
reviews$sent = (reviews$npos - reviews$nneg) / (reviews$npos + reviews$nneg)
reviews$sent[is.na(reviews$sent)] = 0
```

Validating sentiment

The best way to validate dictionary results is to compare them with manual coding. In this case, we can compute the average calculated sentiment per coded sentiment rating:

```
cat(length(reviews$sent))
```

```
## 20473
```

```
cat(length(reviews$overall))
```

```
## 20473
```

```
tapply(reviews$sent, reviews$overall, mean, na.rm=T)
```

```
##         1         2         3         4         5
## 0.1756155 0.3239989 0.3417622 0.4510243 0.4993784
```

So, the higher the sentiment score, the higher the manaully coded sentiment. The correlation between the two is low, though:

```
cor.test(reviews$sent, reviews$overall)
```

```
##
##  Pearson's product-moment correlation
##
## data:  reviews$sent and reviews$overall
## t = 22.347, df = 20471, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.140916 0.167660
## sample estimates:
##       cor
## 0.1543162
```

An alternative is to do linear discriminant analysis with a dichotomous dependent variable, taking only the 5 star ratings:

```
reviews$positive = as.factor(ifelse(reviews$overall == 5, "pos", "neg"))
m = MASS::lda(positive ~ sent, data=reviews, CV=T)
```

And compute the classification accuracy:

```
t = table(reviews$positive, m$class)
sum(diag(t)) / sum(t)
```

```
## [1] 0.6765984
```

Which is not great considering there are only two answer categories.

Applying sentiment to token lists

We can also apply sentiment to a token list, for example the state of the union speeches.

```
library(corpustools)
data(sotu)
sotu.tokens$sent = 0
sotu.tokens$sent[sotu.tokens$word %in% pos_words] = 1
sotu.tokens$sent[sotu.tokens$word %in% neg_words] = -1
head(sotu.tokens)
```

word	sentence	pos	lemma	offset	aid	id	pos1	freq	sent
It	1	PRP	it	0	111541965	1	O	1	0
is	1	VBZ	be	3	111541965	2	V	1	0
our	1	PRP$	we	6	111541965	3	O	1	0
unfinished	1	JJ	unfinished	10	111541965	4	A	1	-1
task	1	NN	task	21	111541965	5	N	1	0
to	1	TO	to	26	111541965	6	?	1	0

And compute the mean sentiment per article:

```
sent = aggregate(sotu.tokens["sent"], sotu.tokens["aid"], mean)
sent = merge(sent, sotu.meta, by.x="aid", by.y="id")
head(sent)
```

aid	sent	medium	headline	date
111541965	0.0600000	Speeches	Barack Obama	2013-02-12
111541995	0.0333333	Speeches	Barack Obama	2013-02-12
111542001	0.0258621	Speeches	Barack Obama	2013-02-12
111542006	0.0606061	Speeches	Barack Obama	2013-02-12
111542013	0.0183486	Speeches	Barack Obama	2013-02-12
111542018	0.0300000	Speeches	Barack Obama	2013-02-12