Introduction to Sentiment Classifiers

- using machine learning methods to pinpoint the sentiment of text
- is this product review positive or negative?
- is this customer email satisfied or disatisfied?
- how are people reacting to our advertisement?
- useful not only for sales - politics/ law making/ sociology/ psychology

Challenges

- Sarcasm, irony, implication hard to define
- change of tone in text
- *"This book was terrible, not because it was bad, but because it was so good: I couldn't put it down until I finished the final pages at 3 in the morning. Not a good thing, when your alarm goes off at 5:50 AM"*

Introduction to Syuzhet

- Four sentiment dictionaries
- Created not only to analyse single sentences, but to explore narrative shifts in books and longer pieces of text
- the following is adapted from Matthew Jockers Syuzhet vignette

Diving into an example

We have a collection of book reviews that we have scraped from our product site.

```
In [5]: library(syuzhet)
        book_reviews <- "This book had me hooked for hours.
            A light read.
            This book was terrible, not because it was bad, but because it was
            I realized, too late to get my fifty cents back, that this is CHICK
            The plot often felt contrived, as pieces fell together too nicely.
            Wow, I'm really torn as to what to say about this book.
            I began reading this book and fell in love with it.
            Reading this book was like an up-hill battle for me.
            This book was a game changer for me.
            Damn if you aren't one of the most problematic things I've ever rea
            Golden has hit pay dirt with this masterpiece.
            I had my doubts about this book, but I'm happy to say I was wrong."
```

get_sentences()

- splits text into sentences and stores sentences in a vector

```
In [13]: sentences <- get_sentences(book_reviews)
         class(sentences)
```

'character'

```
In [16]: sentences[1:3]
```

'This book had me hooked for hours.'   'A light read.'
'This book was terrible, not because it was bad, but because it was so good.'

get_sentiment()

- the get_sentiment function looks for the sentiment of each word or sentence.
- The function takes two arguments: a character vector and a "method", which refers to which of the four sentiment dictionaries to use. The default is the "syuzhet" method.
- Other methods are "bing", "afinn", "nrc" and "stanford"

```
In [18]: get_sentiment(sentences, method = "syuzhet")
         # each output value is an assessment of the sentiment in each sentence
```

0 0 -0.75 -0.25 0.25 0 0.35 0.1 0 -1.75 0.5 -0.5

Each method uses a slightly different scale

```
In [19]: get_sentiment(sentences, method = "bing")
```

0 0 -1 0 -2 1 0 1 0 -2 1 -1

```
In [21]: get_sentiment(sentences, method = "afinn")
```

0 0 -3 0 0 2 3 1 0 -4 1 0

```
In [22]: get_sentiment(sentences, method = "nrc")
```

-1 0 -1 -1 -1 -1 1 0 0 -1 0 0

Short Exercise

-put a favourite quote into the get_sentiment( ) function

```
In [23]: ### your code goes here ###
```

For longer extracts, we can sum the values to

get a measure of the overall feel to a review

```
In [26]: review <- "I think the reason I waited so long to read this series is
I thought it would be too juvenile for my taste.
I was wrong, of course.
I can honestly say that I loved every minute of this.
It's a spectacular little romp with funny, courageous, and endearing cl
It has talking chess pieces, singing hats, a giant three-headed dog nar
tokenized_review <- get_sentences(review)
```

```
In [27]: head(tokenized_review)
```

> 'I think the reason I waited so long to read this series is because I just couldn\'t
> imagine myself enjoying reading about an eleven-year-old boy and his
> adventures at a school of wizardry.'
> 'I thought it would be too juvenile for my taste.'   'I was wrong, of course.'
> 'I can honestly say that I loved every minute of this.'
> 'It\'s a spectacular little romp with funny, courageous, and endearing characters
> that you can't help but love.'
> 'It has talking chess pieces, singing hats, a giant three-headed dog named Fluffy,
> a hilarious giant with a dragon fetish, a master wizard that\'s just a little bit crazy,
> mail carrier owls, goblins running a bank, unicorns, centaurs(!),'

A positive number. Looks like this is quite a positive review.

```
In [29]: sum(get_sentiment(tokenized_review, method = "nrc"))
```

    5

Syuzhet can do more than classify sentences and short excerpts. It can also graph a story's narrative structure, plotting the emotional changes that occur through the text. While there have been some debate about how reliable this the algorithm is, it has been shown to be accurate on a number of literary texts.

```
In [31]: story <- "Poor Harry Potter. After his parents are killed, the lonely
narrative <- get_sentences(story)
narrative <- get_sentiment(narrative)
plot(
    narrative,
    type = "l",
    main = "Harry Potter Plot",
    xlab = "Narrative Time",
    ylab = "Emotional Valence")
```
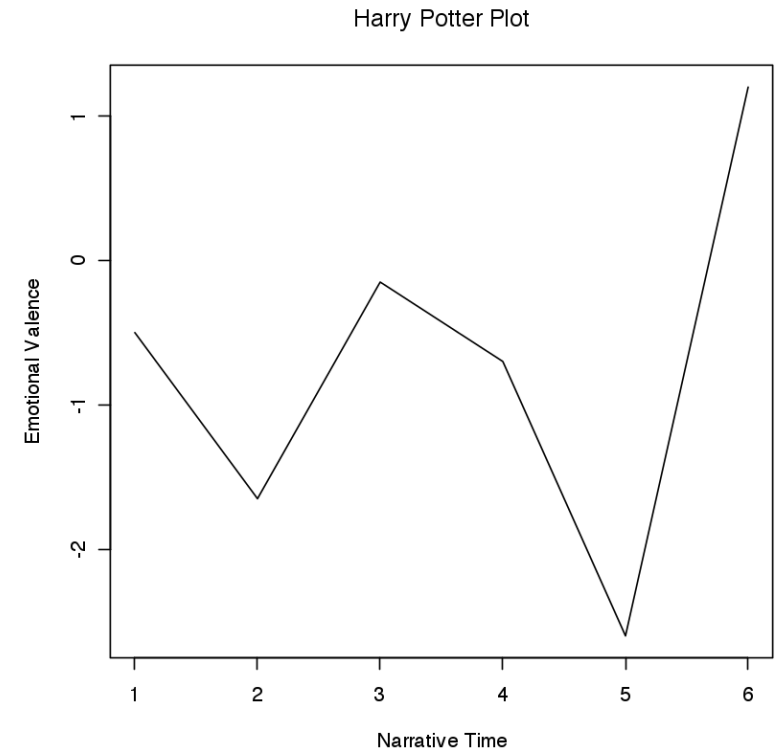

Integrating Sentiment Analysis with Visualisations: How Characters Change in War and Peace

This analysis was posted by Microsoft on their data science blog. Worth a look.

Table of characters and emotional changes throghout the book

	character	BOOK 01	BOOK 02	BOOK 03	BOOK 04	BOOK 05	BOOK 06	BOOK 07	BOOK 08
1	Andrey	-0.8992263	-9.90270886	-7.19243458	-5.6082845	-0.3944617	13.7045850	-2.1104831	-3.6104114
2	Natasha	0.0000000	0.00000000	0.00000000	0.0000000	0.0000000	-0.7016575	0.0000000	1.5553315
3	Pierre	-27.1812298	0.00000000	-2.56592681	-13.8378864	-10.5947793	-6.4762277	0.0000000	-20.3889693
4	Alexander	-1.2489397	0.91031297	2.45537373	0.7663269	2.8567577	1.8528532	0.0000000	0.6627683
5	Alpatych	0.0000000	0.00000000	-4.62947848	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000
6	Anatole	-6.9008031	0.00000000	-1.58288299	-0.6824841	0.0000000	1.1372150	0.0000000	-16.4885800
7	Bagration	0.0000000	-12.05328108	-3.23228025	2.9192063	0.0000000	0.0000000	0.0000000	0.0000000
8	Berg	-0.8274494	0.00000000	6.80684015	-0.5167814	0.0000000	3.5220639	0.0000000	0.0000000
9	Bilibin	0.0000000	4.09909745	2.84215088	0.0000000	2.6358781	0.0000000	0.0000000	0.0000000

Visualise changes through a d3 heatmap

Sentiment about the characters by book