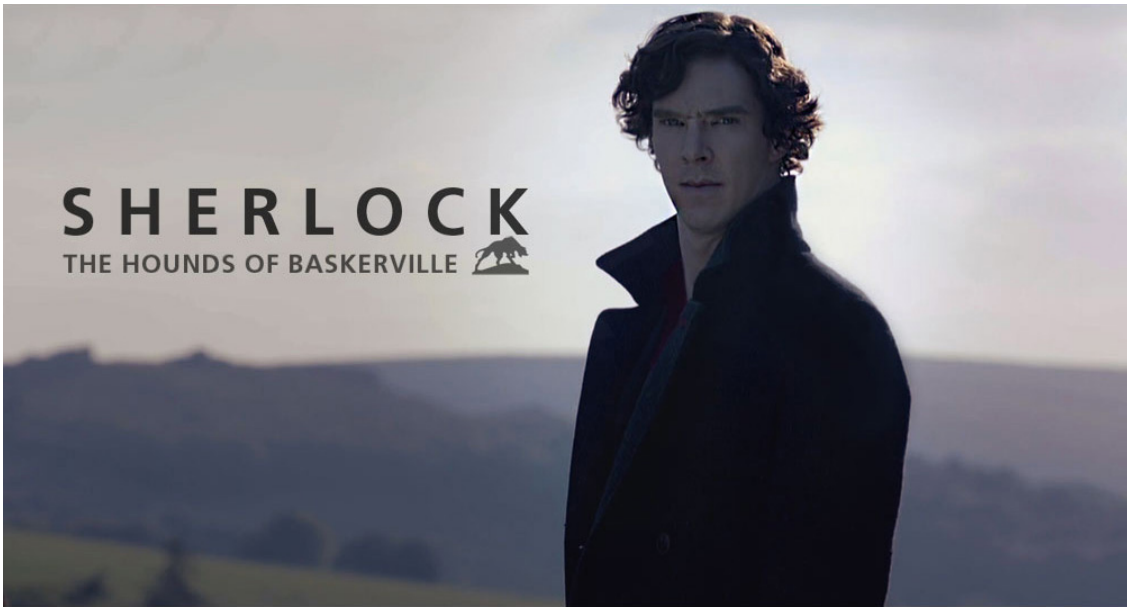
Software Design
Text Mining and Analysis Project

William Lu

February 2015



Overview

Using the pattern.en module by Tom de Smedt, I analyzed a copy of *The Hound of the Baskervilles* by Sir Arthur Conan Doyle for sentiment. The book was downloaded from Project Gutenberg and I was interested in the progression of polarity through the plot of the book.

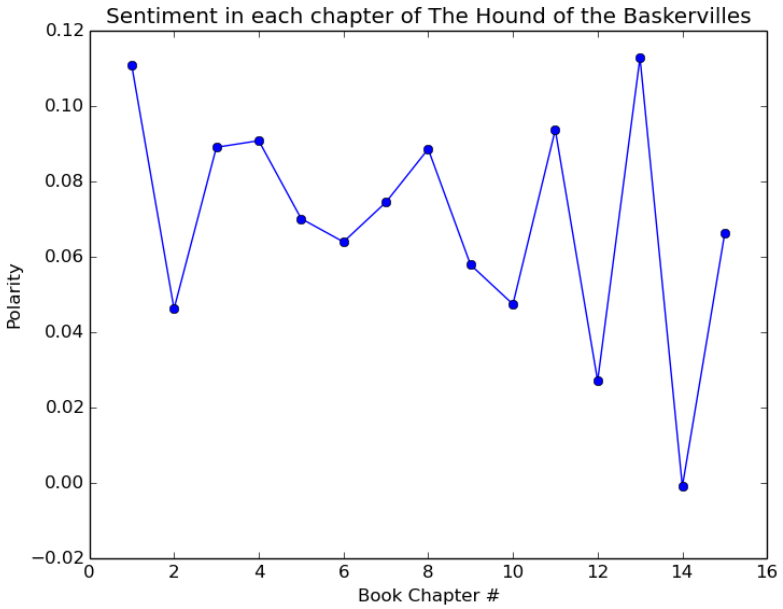
Implementation

To make the plaintext version of *The Hound of the Baskervilles* analyzable, I wrote code in Python that split the text into a list of strings, with each string containing the text for each chapter of the book.

Each string was then fed into the sentiment function of the pattern.en module. The sentiment function generates values for sentiment polarity and subjectivity and outputs them as a tuple of the form (polarity, subjectivity). I made a design decision to not generate a plot including subjectivity data because in analyzing a work of fiction, subjectivity is not applicable and probably inaccurate. The "positivity" of each chapter, however, becomes interesting when you start analyzing its change over chapters.

In order to analyze this change, I used matplotlib, a 2D plotting library created by John Hunter, to plot the data output by the sentiment function.

Results



It turns out that, text analysis of *The Hound of the Baskervilles* for polarity (positivity) doesn't yield extraordinary results. The book is relatively neutral with a slight bias towards positivity (polarity ranges from -1.0 to 1.0). The book starts off quite positively, but drops quickly as the plot of the mystery is introduced and murders are discussed. From there on, the book does not shift dramatically in polarity until the end, where Sherlock fluctuates between seemingly having solved the mystery and distinctly not having solved the mystery.

Something interesting is that the last chapter of the book, where Sherlock recounts his success in solving the mystery, isn't the most "positive." Chapter 13, which seems to be the most "positive" chapter, is actually a sad one, where Sherlock breaks the news of a secret marriage to a character, who becomes shocked and upset. This suggests that the sentiment function isn't necessarily accurate for analyzing the "happiness" of a piece of text. Indeed, this may be the case, as the sentiment function of pattern.en seems to be, from online documentation, more geared towards determining the polarity and subjectivity of product reviews.

Reflection

In the end, I found that my incremental development and compartmentalization of code was extremely helpful in debugging and organization. However, there were some problems with the scope of the project. I began coding with the intention to analyze many books, but realized very quickly that no one book was formatted exactly the same way - each book had different headers before the actual text (e.g. some had tables of contents and some did not) and many books had different ways of denoting the beginning of chapters. There are ways to deal with this problem, but would require significantly more time and energy than the intended scope of this project.

My plan for unit testing was improvised, as doctests were not as flexible as I wanted my unit testing to be. I printed out variables at various stages during the development process, and if what I printed looked correct (i.e. a list of strings where there was supposed to be a list of strings), I knew my code was working. Going forward, I will try to find easier ways to unit test my code, and will try to further find ways to make my code easier to debug.