Using Natural Language Processing to Analyse the Shape of Stories

Luca Davies
BSc. (Hons.) Computer Science

March 2020

TO DO LIST

- Plus-minus symbol HOW???
- Fill out Introduction more
- Fill out background (with references)
- Anymore to add to Design/Dev section?
- Chapter dividing???
- Redo Cinderella hand analysis + write up anything interesting
- Tidy Reader Reflection
- More on Early Modern vs Modern English
  - Reference Jonathan Culpeper
- Comedy vs Tragedies
- Conclusion
  - Summary
  - Relfetion
- Project Gutenberg INCONSISTENT HTML FORMAT

Declaration

I certify that the material contained in this dissertation is my own work and does not contain unreferenced or unacknowledged material. I also warrant that the above statement applies to the implementation of the project and all associated documentation. Regarding the electronically submitted work, I consent to this being stored electronically and copied for assessment purposes, including the School's use of plagiarism detection systems in order to check the integrity of assessed work.
I agree to my dissertation being placed in the public domain, with my name explicitly included as the author of the work.

Name:
Date:

Abstract

This report examines the application of Natural Language Processing and Sentiment Analysis to fictional texts in an attempt to summarise narrative arcs as a curve on axes of time against positive/negative sentiment. VADER is used to process text for sentiment analysis and further experimentation is carried out to analyse how suitable VADER may be for this task. Corpora was mainly sourced from Project Gutenberg. A tool was developed to carry out a range of experiments on sentiment analysis that employed VADER as it's sentiment analysis engine. Experimentation was carried out in the context of sentiment analysis around the following topics: the coarseness of sentiment analysis; the accurarcy of VADER at sentence level via hand-tagging; sentiment analysis vs. analysis by human readers; early-modern English vs. modern English; Shakespearean comedies vs. tragedies. It was discovered that a more powerful and versatile sentiment analysis engine would likely be needed to produce more readable curves and that a more flexible tool would prove useful for processing different text styles, such as scripts.

Contents

1 Introduction

1.1 Overview

Writer Kurt Vonnegut suggested at various points in his career that all stories may be categorised into a relatively small number of basic archetypes based upon the emotional ups and downs experienced within the narrative. He gave each of these novel names such as Man-In-Hole and Boy-Meets-Girl as a simple reference for each. [3]

Literature is one of the defining features of the human race. No other creature on Earth can writes, let alone writes about itself. Humanity takes this a step further again, making up fictitious stories, some rooted in reality, some with no basis at all in which we dream and imagine worlds and situations that may never be possible to achieve. Through written word we express emotion. All emotions, happiness, sadness, anger and fear; love, hate, awe and grief – everything humans can feel, we write about. We create characters to express and receive these emotions, they act as vehicles to transfer their feelings to a reader. With all this freedom to write and create without bound, are we actually as free as we perceive?

Vonnegut suggests that all stories are members of one of a very small number of categories of story that define roughly how the emotional progression of that story pans out. Do writers naturally and unknowingly write literature that falls into these types or is each story as unique as the next, taking its readers along its own path as they go?

This concept and these questions drew me toward this project proposal. Not only is it a fascinating endeavour to map the emotions of a novel, but as an exploration of the freedom of writers to convey emotions and of how humans often generate categories and groups without even trying.

With the high-level view clarified, it's important to note the tools within natural language processing that are relevant, namely that in this study, sentiment analysis takes the spotlight. Emotional analysis is a more in-depth field that shifts focus toward the psychological analysis ([1]) and employs machine learning and artificial intelligence to further predict and understand the emotions presented in text.

1.2 Motivation

At it's core, this study is novel. It is interest driven - to attempt to show that stories can be categorised in a very simple and easy manner based upon the emotional arcs they lead readers across is an interesting new way to sort fiction. There may not be any exlicit *need* to categorise literature in this way, however it has the potential to lead to further studies that examine just how it is that humans create literature and nay patterns we follow.

1.3 Aims & Objectives

The aims of this report are as such:

- Design and develop an application to process a range of corpora to produce a graphing of the emotional arc during its literary course (as produced using SA)
- Analyse a range of corpora for compliance with Kurt Vonnegut's theories and story shapes
- Otherwise attempt to identify potential trends in the texts processed, such as obvious geometric differences between literature generally considered happy/sad
- Present graphs of texts to readers who are familiar with the text to assess if their perceptions of the text align with the SA graphs
- Assess VADER's ability to process text outside its design remit (e.g. early-modern English)

1.4 Report Structure

The reminader of this report will disucss relevant background and context, the design and implementation of the Sentiplot tool, followed by a detailing of the experimentation carried out. The report will then be concluded by an analysis of the results acquired from this experimentation.

2 Background

2.1 Overview

This chapter will examine and summarise existing literature and studies in this area and topic. That is, natural language processing and sentiment analysis as a tool for extracting statistical data that describe emotions in (mainly) fictional works of literature.

The processes involved for finding useful information included searching for online articles and pre-existing projects while searching libraries to use in the implementation, Google Scholar, Lancaster University Library OneSearch and Kurt Vonnegut's own lectures on this topic.

Existing projects will help to prove the developed application is performing up to standard (or not) and can be used as a side-by-side comparison of literature processed for this report and by these projects as well as to examine the processing of literature unavailable for the purposes of this report.

2.2 Natural Language Processing & Sentiment Analysis

Natural Language Processing (NLP) is a very broad field concerned with, at a high level, the understanding of human language by computers. It fuses linguistics with computer science to not just parse but to *understand* human language, to understand it in such a way that meaning can be deduced and emotion and sentiment can be extracted, even when not fully clear.

Via the beginnings of machine translation, NLP has existed as a research field for decades, even before the current name was coined. Breakthrough projects like ELIZA made progress in the field, but it wasn't until the 1980s that statistical methods began to be used. Prior to this, large sets of hand-written rules governed how NLP systems worked, which, although sometimes effective, limited the overall progress to the manual effort put into the models and rules. In modern NLP toolkits, models are still prevalent, but instead of them being assembled by hand, they are often formed using machine learning techniques based upon employing training data, test data and then real language data.

Sentiment Analysis (SA) is a subfield of NLP that focuses on extracting and subsequently quantifying opinion and sentiment around a topic simply by analysing text. SA is a particularly new field, and has so far mainly be used on corpora that is factual and descriptive: product/movie reviews and social media to gauge general public opinion.

2.3 Kurt Vonnegut on "The Shape of Stories"

Vonnegut described a number of potential story types as displayed below in figure something. He suggested that all stories fit into a very small number of categories, and moreover, that stories from different cultures around the world may generally trend toward different story types compared to elsewhere. In his lecture on the topic Vonnegut draws out the curves of some well known novels and stories to demonstrate his meaning. Drawing distinct curves with very defined turning points and changes in direction that match up with points in the given literature.

2.4 The Hedonometer Project

The Hedonometer project was established to gauge happiness, the world over, starting with Twitter and other social media outlets. The project's scope has since widened to process corpora direct from books, film scripts, news outlets and other foreign language literature.

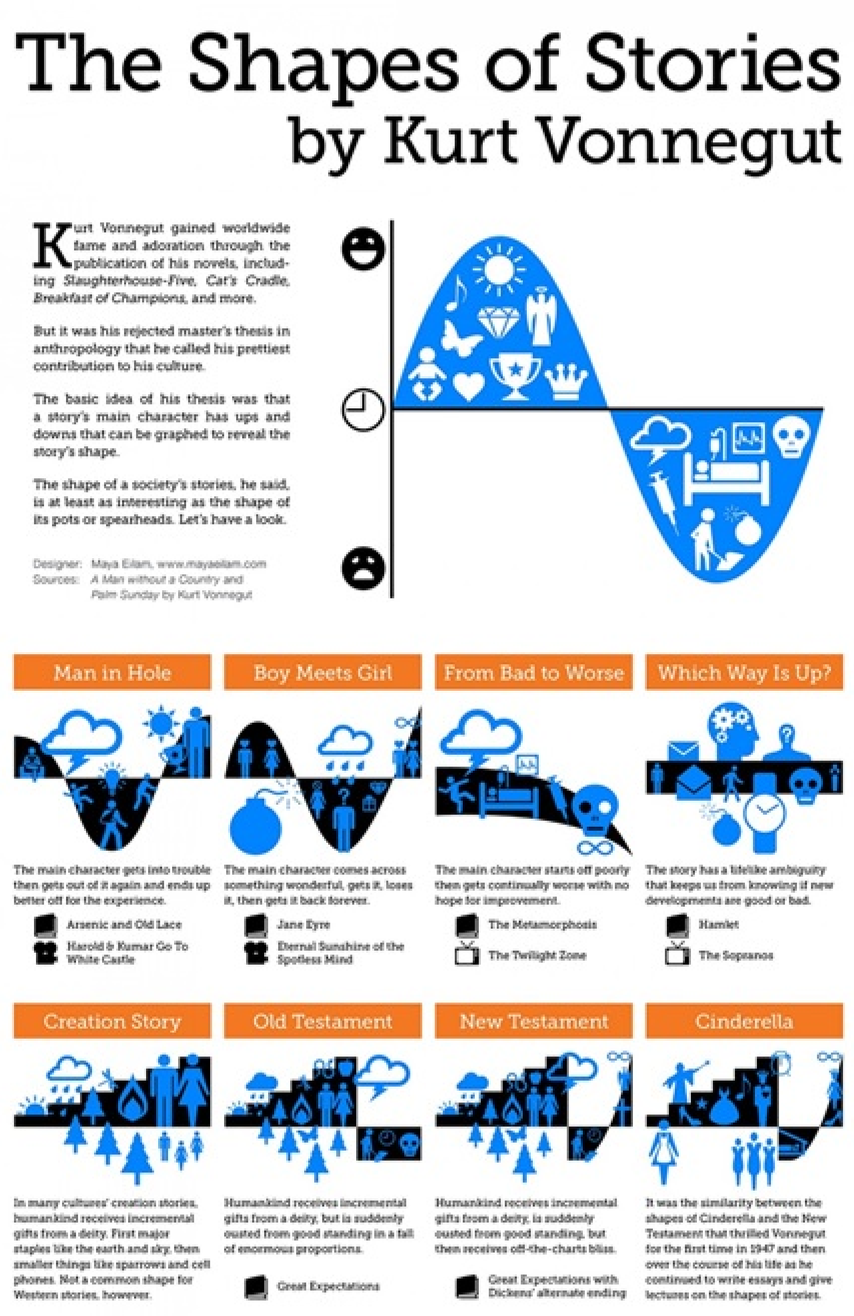


Figure 1: Vonnegut's Story Types

3 Sentiplot Tool

This section will detail the design and implementation of the tool used to carry out this study, *Sentiplot*.

3.1 Languages & Libraries

This section briefly discusses the choice of programming language and NLP/SA tools used in the Sentiplot tool.

3.1.1 Top-level Language

When selecting which language would be most appropriate to use for this project, my major considerations were my prior knowledge of potential languages and their ease of creation of a pleasant user interface. Availability of NLP tools in each language was also to be considered, but as the end implementation shows, this was not imperative due to cross-platform and cross-language capabilities of the chosen combination of languages and tools.

A listing of considered languages follows:

- Java
  - Strong language knowledge and familiarity
  - JavaFX and Swing available for interfaces
  - Native language of Stanford CoreNLP
- Python
  - Very limited language knowledge
  - No knowledge of interface building
  - Native language of industry standard NLTK
- C# (with .NET)
  - Strongest language knowledge and familiarity
  - Extreme ease in creating interfaces via Windows Form using Visual Studio
  - Cross-platform variants of CoreNLP and VADER available via simple packages

Taking all these points into consideration C# was selected, due to its pros specific to myself and the availability of non-native libraries via APIs. This permitted me to use the full Microsoft Visual Studio suite for development, including the Windows Forms designer.

3.1.2 Natural Language Processing Tools

Various tools across a wide array of languages are freely available to provide standard NLP functions and advanced processing capabilities.

Initially a full CoreNLP pipeline was employed to process corpora but this proved to be extremely slow, loading around 2 gigabytes of models very slowly into memory before even processing anything. The pipeline was modified to tokenise and sentence-split the input only, and the actual sentiment analysis was performed by VADER.

Both the CoreNLP and VADER libraries are non-native to C# but have easy to use APIs for direct manipulation of their types and methods outside of their native environments. CoreNLP has an in-house developed API for C# and VADER has a third-party API called *VADERSharp*.

3.2 Design & Development

Sentiplot was developed over the course of two academic terms between early October 2019 and late January 2020. The first few weeks involved mainly research into what language was to be used and what NLP/SA tools or toolkits would provide the best mix of suitability-to-task and ease of use. The initial framework and basic features were developed using a waterfall type methodology before later, more experimental features were developed in a more ad hoc manner using an agile development process *after* the main program with built.

The base applicaition was projected to take 4-5 weeks to develop from scratch and the smaller experimental features added after to each take aroudn a week. In reality, this first stage of developement occupied the majority of the time from the start of November through till mid December. Then each of the experimental features pushed on frm there, rolling into the next term, rather than all being finished before the Christmas holidays. Thankfully this did not cause major issues as there was an excess of time set aside for write up in the second term. Final bits of development including housekeeping and codestyling wrapped up approaching week two of February 2020, still allowing time for this report write-up.

As in discussed in section 3.1.1, C# was the language used and Windows Forms was the user interface framework chosen. A full Visual Studio (Community) development suite was used to design, build and test the application which allowed for relatively switft and easy development due to the neat integration bewteen the technologies. Had a different, less familair language/framework been adopted, further schedule overruns may have occured. StanfordCoreNLP was at one point the NLP tool of choice, however, dificulty and delay in marrying its Java libraries to C# and setting up its processing pipeline are in large part what encouraged the eventual swap to VADER in early December.

As the language and framworks use lend themselves to it, the application is built with an object-oriented pattern in mind, however this doesn't come to light much as the application is somewhat self-contained - no other classes or applications need interface with it, Sentiplot only uses simple API calls to any external libraries.

The overall internal structure of Sentiplot is not massively complicated. It is detailed in section 3.3.

3.3 Implementation

3.3.1 Structure

The application is written in C# .NET interfacing with both Java and Python libraries via APIs as detailed in sections 3.1.1 and 3.2. Windows Forms was chosen for the interface as a quick and easy to build platform, stable on Windows with seamless integration into C# and the .NET framework.

The application is composed of two forms, each with their designer code. The first allows the user to select a file to load text from and set processing granularity. The second presents the results of the analysis in multiple ways and provides facility to save these results as images of the output graphs.

3.3.2 Sentiplot Form (Main Form)

This is the main form for the application. It is loaded on start-up and contains all needed information or otherwise calls other classes to complete its task.

Its key functions include:

- Initialising the CoreNLP pipeline ot tokenise the input text
- Generating an OpenFileDialog object to allow the user to select a *.txt* or *.html* file
- Parsing the content of the input file to prepare it for processing using regular expressions and simple character-based splitting

3.3.3 ResultsViewer

This form displays the results of VADER's processing of the input text. It displays a graph of the entire text from start to finish, a full list of all tokenised sentences and their associated sentiment scores (in tabular form) and individual graphs for each chapter in the text (HTML file only).

- Handling the output data and feeding it into the main chart and table
- Allow hiding/showing of each different type of sentiment score for the main graph
- Dynamically producing more tabs on the form to show each successive group of five chapters

3.4 User Interface

The implemented interface did not need to be excessively complex or particularly appealing as it mainly served to function as a harness to conduct the study, with more importance placed on the code-behind and results.

To provide quick results and ease of programming, I used the Windows Forms suite to produce a visually simple but functional interface. It has two screens: a screen to load the desired text to be processed and to set the granularity of the analysis, and a screen to present the processed results. The following sections provide an overview of these screens.

3.4.1 Text Selection & Options

Figure 2 shows the main presented screen, after having loaded an HTML file (Hamlet in this case) which has been parsed to produce plaintext, having stripped out all the unneeded HTML tags.

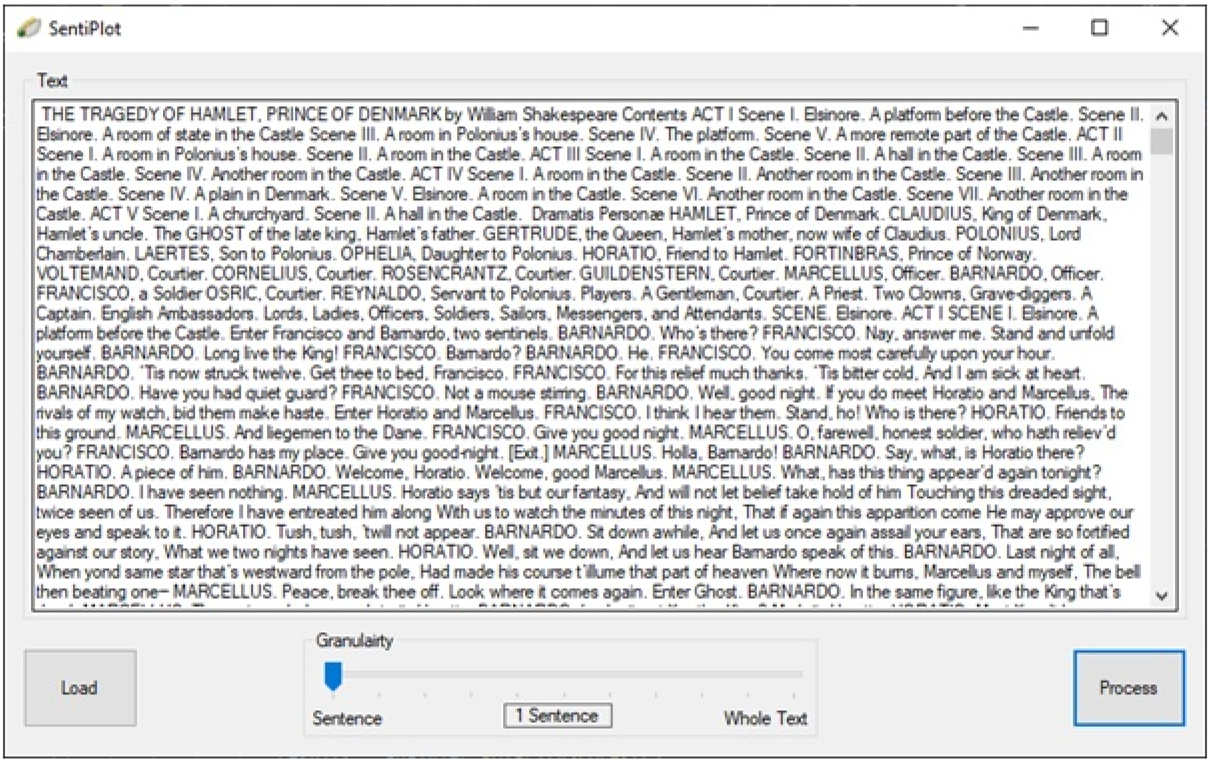


Figure 2: Main Sentiplot window

3.4.2 Results Display

Figure 3 shows the results screen for Hamlet, initially showing the graph of the entire text. The maximum and minimum points are labeled with the start of the related sentence. Hovering the mouse over these labels shows the entire sentence or sentences that produced that datapoint.

For every sentence analysed VADER returns four sentiment scores: positive, neutral and negative (each holding a -1 to 1 score regarding match to that sentiment), and compound which acts as a single representation for the sentiment in the sentence parsed. Each of these scores have their own graph which may have their visibility toggled on and off with the check boxes in the bottom left. (Default is compound alone, as it proved the most indictive of sentiment, as advised by VADER documentation.) This graph can be saved to a JPG by clicking the save button.

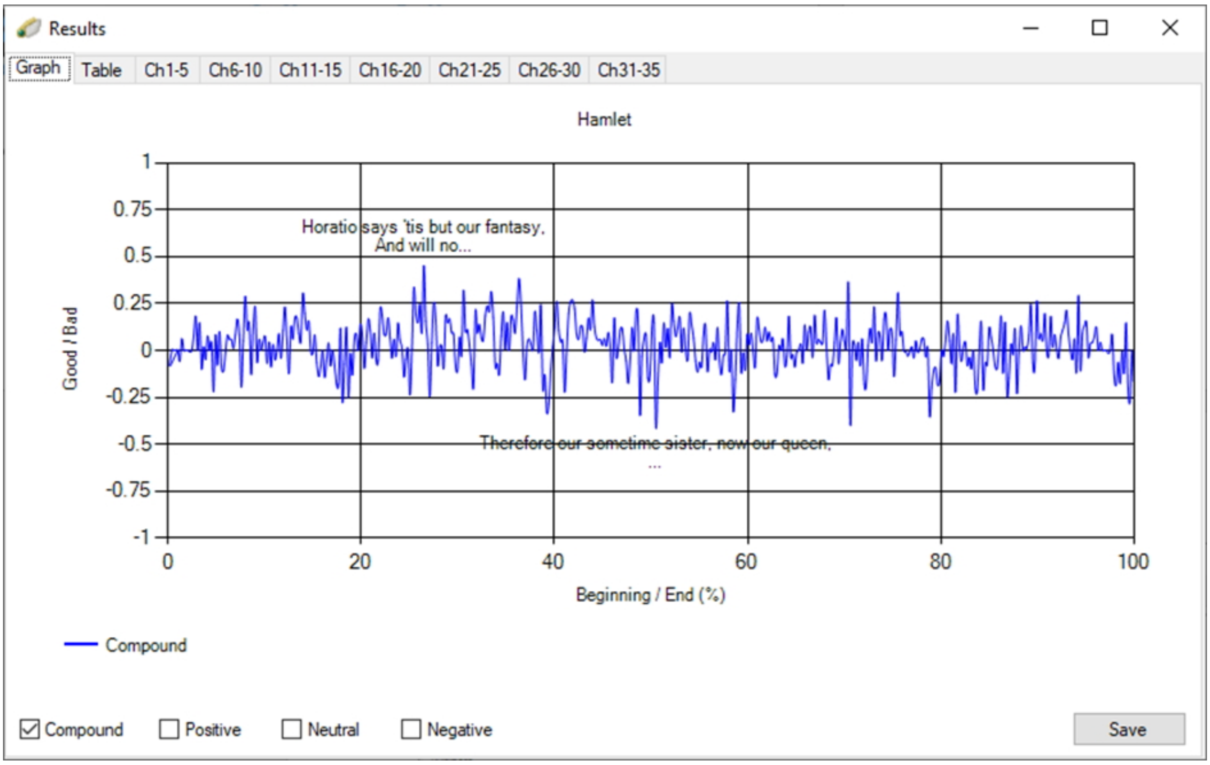
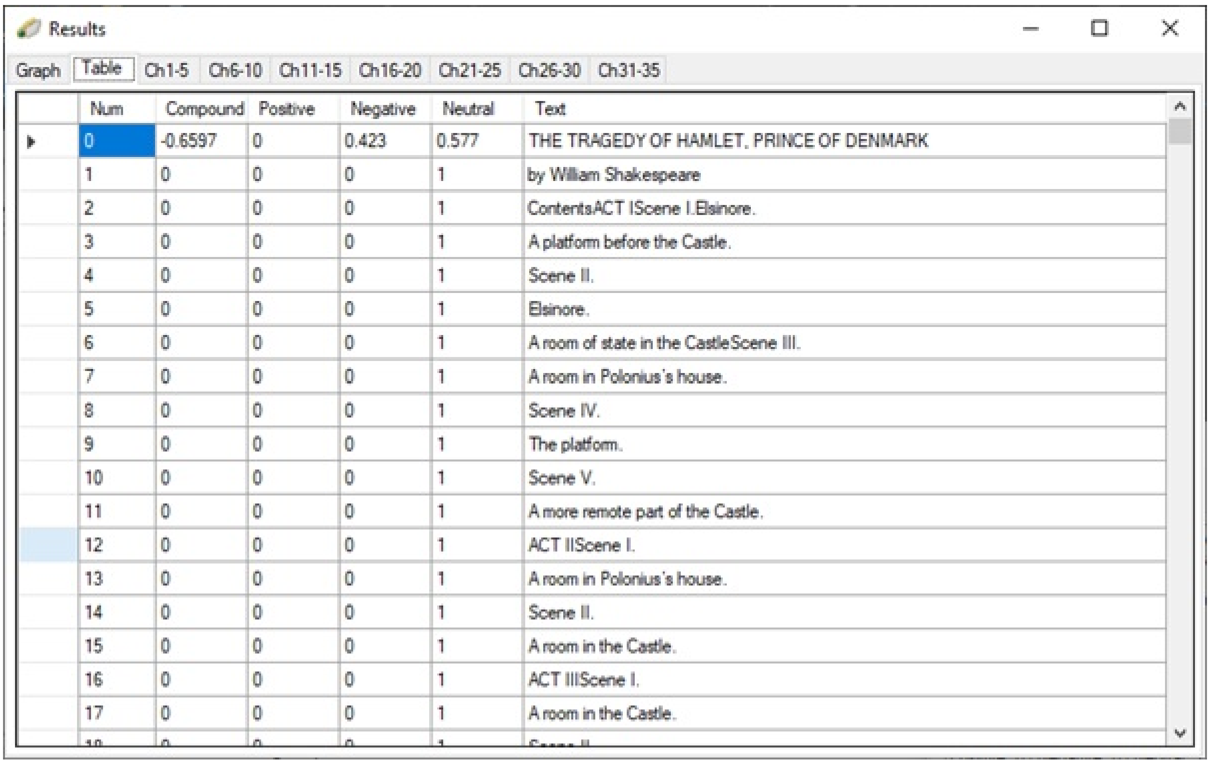


Figure 3: ResultsViewer window

Figure 4 shows the Table tab of the results screen. This simply lists each individual sentence token in the input with VADER's output value for the four scores mentioned previously. The table's default ordering in by sentence index, or the chronological order in the text, but the it may be order by any of the fields shown.



Num	Compound	Positive	Negative	Neutral	Text
0	-0.6597	0	0.423	0.577	THE TRAGEDY OF HAMLET, PRINCE OF DENMARK
1	0	0	0	1	by William Shakespeare
2	0	0	0	1	ContentsACT IScene I.Elsinore.
3	0	0	0	1	A platform before the Castle.
4	0	0	0	1	Scene II.
5	0	0	0	1	Elsinore.
6	0	0	0	1	A room of state in the CastleScene III.
7	0	0	0	1	A room in Polonius's house.
8	0	0	0	1	Scene IV.
9	0	0	0	1	The platform.
10	0	0	0	1	Scene V.
11	0	0	0	1	A more remote part of the Castle.
12	0	0	0	1	ACT IIScene I.
13	0	0	0	1	A room in Polonius's house.
14	0	0	0	1	Scene II.
15	0	0	0	1	A room in the Castle.
16	0	0	0	1	ACT IIIScene I.
17	0	0	0	1	A room in the Castle.

Figure 4: ResultsViewer window showing table

Figure 5 shows one of the chapter tabs from the results screen. Each of graphs the compound sentiment score of up to 5 chapters from the processed text. Again, maxima and minima are labeled with their relevant sentence(s), expandable by hovering the mouse over the label.

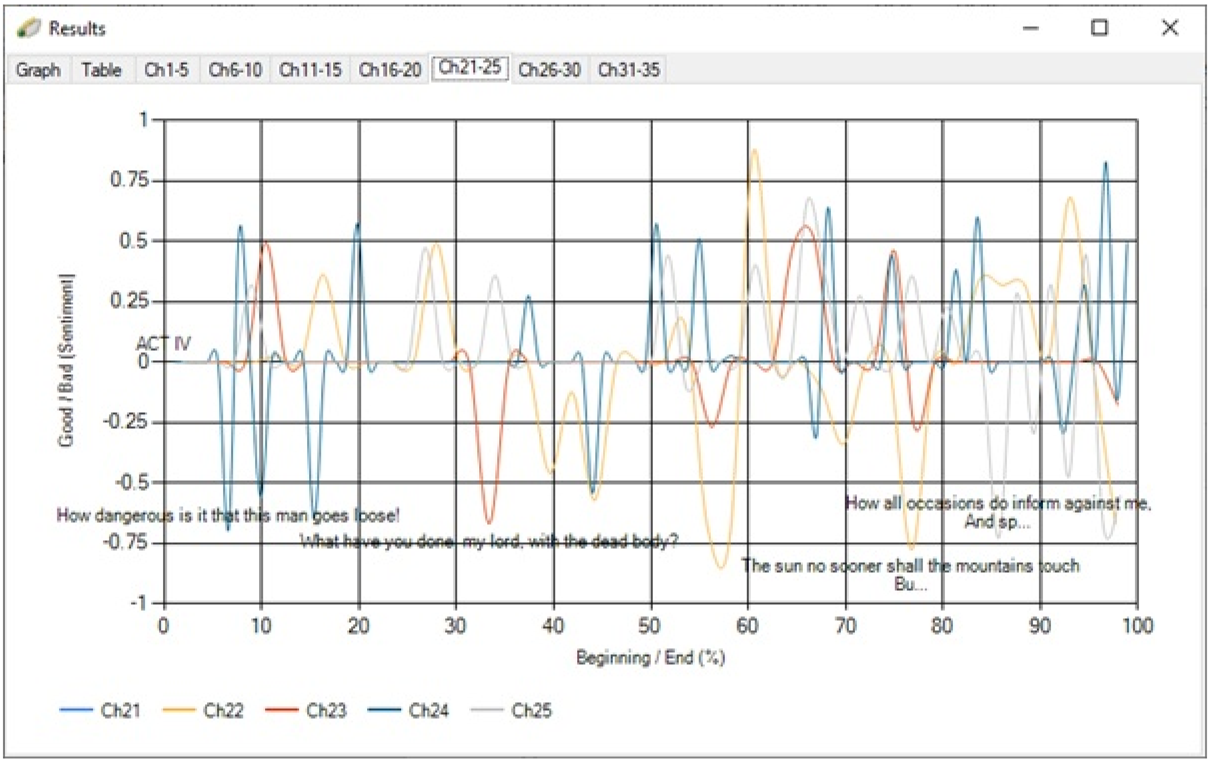


Figure 5: ResultsViewer window showing chapter graphs

4 Experimentation

The following section covers the various experimentation and analysis carried out using Sentiplot as a tool for Sentiment Analysis and graphing accordingly.

4.1 Analysis Block Size

Sentiplot incorporates one primary setting for its analysis of texts: the granularity slider. This allows the user to select varying degrees of granularity for the produced graphs (as a percentage of the text being analysed). This allowed analysis of texts at a number detail levels to find which produced the most obvious curves to conform with Vonnegut's story curves. This may have introduced a small amount of positive-results bias, accepting those granularities that appear most curve-like for a given text instead of what may have actually reflected the curve of the text.

The options provided and their effectiveness are shown in figure 6.

Too Fine	**Acceptable**	**Too Coarse**
1 Sentence	0.2%	5%
0.1%	0.5%	10%
	1%	25%
	2%	50%
		100%

Figure 6: Various granularities are available for processing text, this table shows those that proved the most effective and those that did not.

This set of experiments varied the analysis blocksize for a given text and compared the output graphs for the entire text with a view to identifying a specific curve for the text. The comparison is qualitative, employing only human visual reference as opposed to any mathematical function. The purpose of this part of the study was as a forerunner to further analysis and experimentation to find the best setting or the best range of settings for either specific or generically for all texts to produce a coherent sentiment curve. Figure 7 show the differing graphs that may be produced by changing the granularity.

VADER is designed for parsing short lengths of text (less than 140 characters, as per a Tweet), so in order to produce more coarse graphs than single-sentence, VADER is still fed text on a per-sentence basis with the results summed up to N sentences where N is the corresponding number of sentences for a given granularity setting for a given text. The relevant data point is then plotted N sentences beyond the previous (i.e. at the end of the analysed block).

By default, every sentence of the text is plotted as its own point. This gives highly detailed graph but it very difficult to discern any sort of shape. Sentence to sentence, sentiment can vary wildly from the most negative in one to the most positive in the next (context and corpus dependent). Due to this, the resultant graph is highly spiked and difficult to garner information from.

By contrast, using the more coarse options of 25% and upward produce graphs that do show a clear line, the hope being the larger regions that have an average higher/lower sentiment score can then be picked out. However, at this level of coarseness, the data is compressed so much so that most of the detail is lost, smoothing the curve beyond useful limits for purposes of analysis

Through trial-and-error experimentation, the ideal setting was found to lie between 0.2% and 2%, erring toward the 2% for most texts. Although the choice to use percentages for the granularity setting was made to better cater to both long and short texts, those that are particularly long (¿5000 sentences) and those that are particularly short (¡200 sentences) may require different settings to best show the data and thus the curve. This is due to the fact that a longer novel does not necessarily have longer chapters and/or longer sections that could be identified as having a partiular bias in sentiment.

With all this being said, it's possible, that a different tool may have been better suited overall to this task than VADER is, due, at least in part, to it's native design choice to parse text the length of Tweets, not entire corpora. Sentiplot could be redesigned to pass VADER each block as a whole instead of summing the individual sentence values instead, although it is to be assumed that this would yield less accurate results as this is not how VADER was intended to be used.

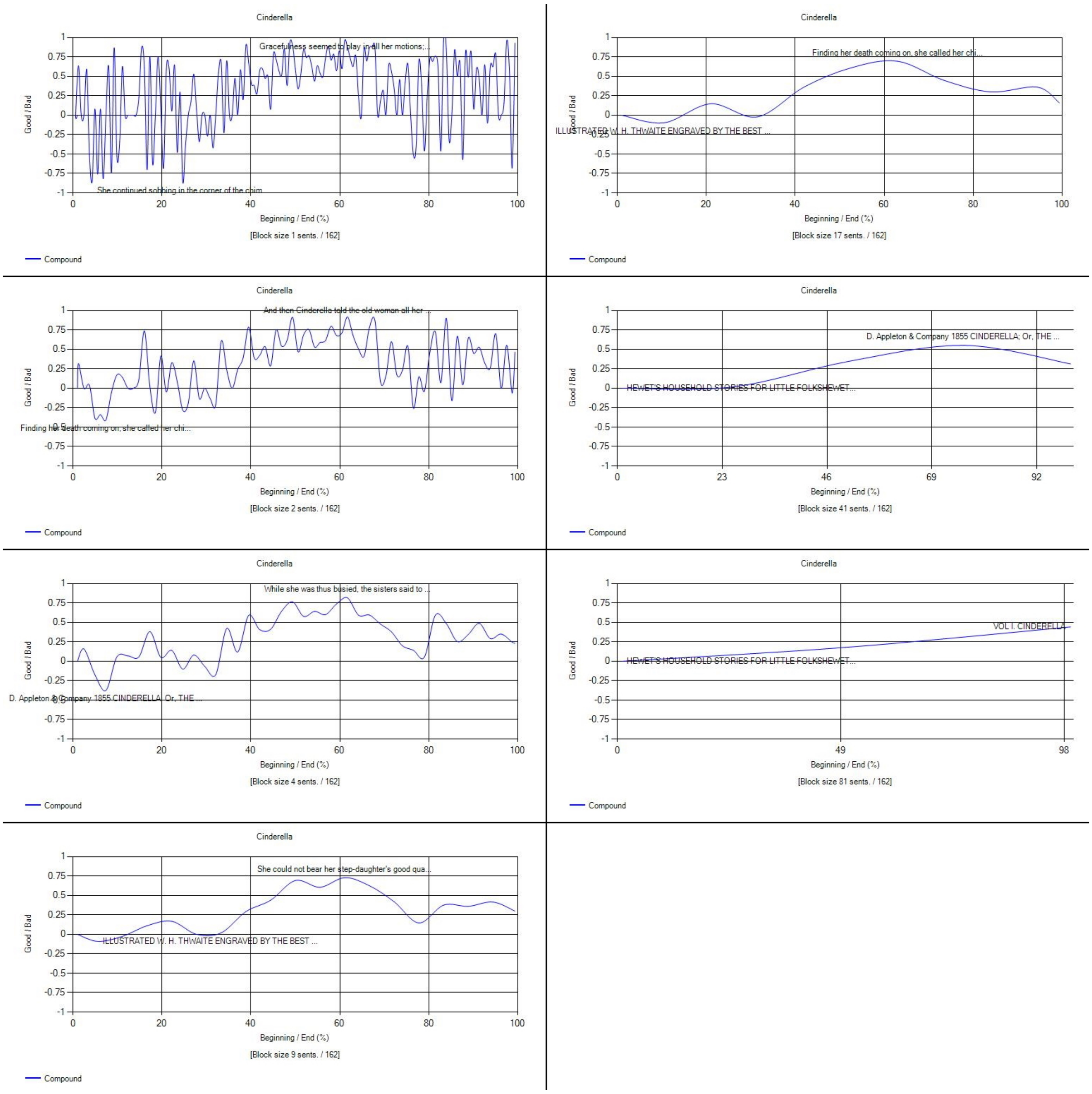


Figure 7: Each graph shows the result of decreasing the granularity of analysis and show how there appears to be a sweet spot. In this case, four of the highest options summed only a single sentence due to short length of the text. Only one of these has been included.

4.2 Curve Identification

The original inspiration behind this project stemmed from Kurt Vonnegut's lecture on the "The Shape of Stories" and atteming to test his theory by plotting the sentiment results of various texts on a graph against time. This is what drove the initial development of Sentiplot and this study as a whole. However after inital results were produced by Sentiplot, one of two things became somewhat clear: eiter Vonnegut's theories were wrong, or VADER was not up to the task at hand. Which, remains to be seen and is likely best left to further research. Nevertheless, this meant the aims of the project began to shift more toward NLP for processing literature in general (as opposed to its normal domain of opinion mining) and analysing those results. Consequently potential developements for curve identification were not developed to completion, namely, an algorithm to statistically compare the produced graphs with those predicted by Vonnegut, was never written. Despite this, mathematical plots were extraced from the six distinct curves mentioned in figure 1 (exlcuding 'Which Way Is Up?' and 'New Testament' for the fact of being a flat line and identical to 'Cinderella', respecitvely) by using an online point plotting tool (see [2]). The raw CSV data is provided in the additional documents to this report.

The following sections breifly explore the potential loose identifications of curves in a number of texts processed by the Sentiplot tool.

4.2.1 Whole Text Analysis

A small variety of both novels and scripts of varying legnths were processed to produce their curves. As mentioned above, there was not an immediately obvious correlations between many of the texts' graphs and those predicted by Vonnegut. *Great Expectations*, *Metamorphosis* and *Cinderella* were the three 'novels' processed in decreasing order of length. *Great Expectation* was the longest text processed with just shy of 10,000 sentence, however the only curve it could be sit to fit at face value is 'Which Way Is Up?', or a flat line. There's variation, but only minor. Analysis of the curve from a larger perspective, one could argue it does *almost* fit Vonnegut's predicted 'New Testament' curve, albeit with an extremely reduced y-axis scale, and loss of all complex features of the graph.

Kafka's *Metamorphosis* shows a simialr situation: the prediction is 'From Bad To Worse', an ever decreasing parabola of sorts, however Sentiplot's output is at closest match, neutral, but at worst seemsto have a few peaks and troughs of particularly negaitve *and* positve sentiment.

Cinderella is the one graph that acutally does bear somewhat of a resemblance to Vonnegut's prediction, being not too far removed from the rises and falls of his 'Cinderella' curve, if slightly less extreme. See figure 8 for the produced graph overlaid with a representation of Vonnegut's curve transformed to fit the axes. The lack of of the harsh drop at 70% may also be explained by the fact that block size averaging smooths out harsh changes such as this.

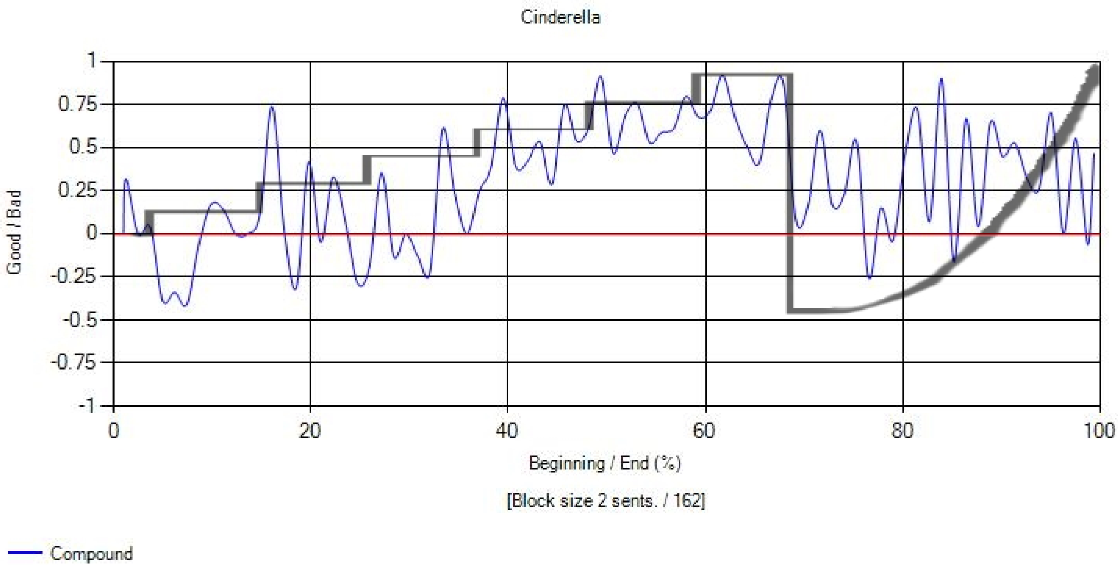


Figure 8: The curve produced for Cinderella matches smoewhat closely with Vonnegut's prediction.

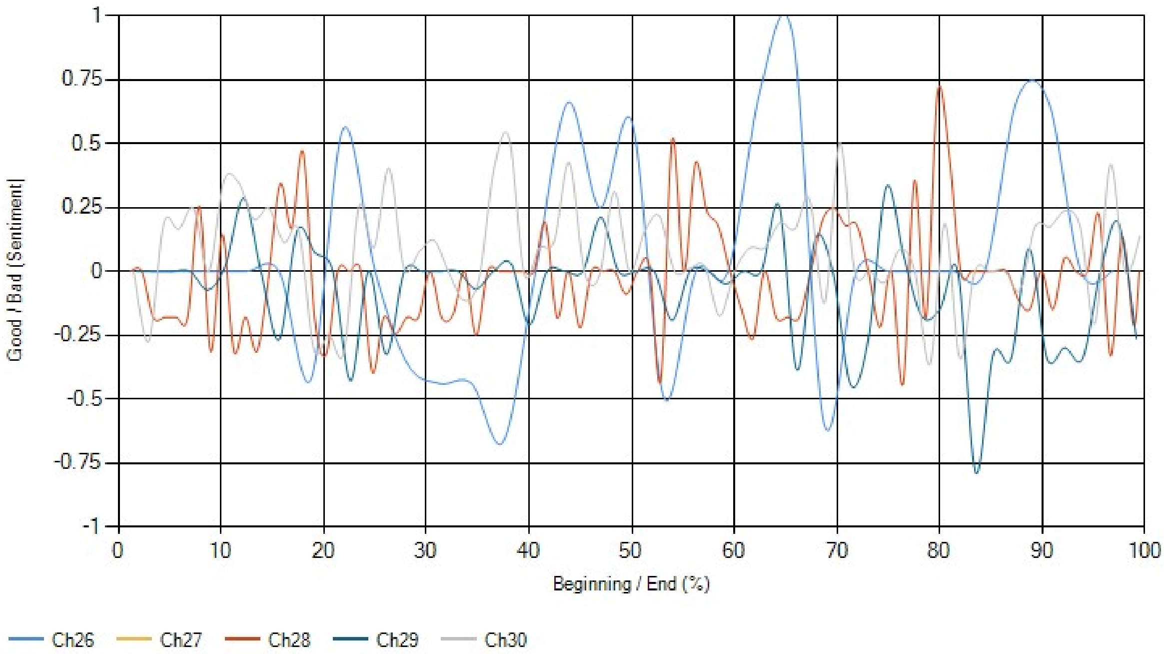


Figure 9: "Chapters" 26 to 30 of Macbeth showing five overlayed curves. Chapter 26 (light blue) shows potential shape albeit somewhat extreme.

4.2.2 Chapter Analysis

This functionality was implemented in Sentiplot to further examine not only if stories had curves at all, but to see if any such curves could be discerned within smaller narrative sections of a larger text. The method is not greatly complex, dividing a text into chapters and simply running and graphing and plotting each chapter individually to ease analysis.

Each chapter is plotted using the same granularity as the main text, this being a percentage of the input text length. Here, this percentage is applied to the length of the chapter itself. Depending on average chapter length this can mean that specific a granularity many have to be chosen to examine chapters. Being much shorter than the full text, the 1% often used for a whole text can lead to single-sentence processing for many chapters.

4.3 Hand Analysis Vs. VADER

This test was performed by selecting a 100 random sentences from two texts (Macbeth and Cinderella), analysing them each individually by hand and estimating a combined positive/negative sentiment score in the same range as that produced by VADER. I attempted to avoid trying to emulate the way VADER would process the text, instead scoring each sentence as I would as a reader of the text: "Would I feel positively and negatively after reading this sentence?" or, similar - a human approach.

The motivation behind this was to understand if VADER's output results were trustful, totally incorrect, or somewhere in the middle. It's easy to blindly follow what a program says but NLP and Sentiment Analysis is still very much a field that cannot yet directly emulate the processes of the human brain.

The results obtained, although they are based on one person's own judgement (and hence are not necessarily reliable), were disappointing to say the least. To determine similarity, a threshold difference of +/-0.25 was chosen to mark the boundary between results that correlated with each other, and results did not. From this, of the 100 sentences taken from Cinderella, only 27% matched. In the case of Macbeth, 50% matched, but it is possible to theorise that majority of the increase is likely due to the fact of Macbeth being a script - namely, there were many sentences that were just a name, or just a stage direction alone and thus was very easy to be rated neutral (or, zero) by both a human and VADER. However, in the 100 sentences, 29 names, stage directions or similar were present, of which 10 matched (mainly at or near zero), decreasing the effective percentage of matched sentences to somewhere in the range 21%-40%. There were some cases where a plain name resulted in an extreme score (see rows 4 and 9

Sentence	Human Score	VADER Score	Difference
Let grief Convert to anger; blunt not the heart, enrage it.	-0.85	-0.6093	0.24
I'll go no more: I am afraid to think what I have done; ...	-0.7	-0.296	0.40
Let not your ears despise my tongue for ever, Which ...	-0.2	0	0.20
MACBETH.	0	-0.6597	0.66
SCENE II.	0	0	0.00
Well, well, well.	0	0	0.00
I grant him bloody, Luxurious, avaricious, false, ...	-0.9	-0.0772	0.82
or is it a fee-grief Due to some single breast?	-0.4	0	0.40
LADY MACBETH.	0	0.7096	0.71
Thou canst not say I did it.	-0.05	0	0.05
Help me hence, ho!	0	0	0.00
ROSS.	0	0	0.00
Macbeth shall sleep no more!"	-0.1	0	0.10
—A seventh!	0	0.5073	0.51
[Exit.]	0	0	0.00
List'ning their fear, I could not say "Amen," When they ...	-0.3	0	0.30
I dare not speak much further: But cruel are the times, ...	-0.8	0	0.80
or why Upon this blasted heath you stop our way With ...	-0.25	0	0.25
Woe, alas!	-0.1	0	0.10
MACBETH.	0	0	0.00

Figure 10: An excerpt from the table of human-analysed sentiment scores along side VADER's scores with the difference (highlighted green for a match, red otherwise) shown. The original sentences from Macbeth are shown for reference.

in figure 10). Some of the scores assigned by VADER also seem to make little to no sesne whatsoever, for example, row 7 in figure 10: the full sentence directly includes (without negation), the words "bloody", "avaricious", "false", "deceitful", "malicious", and "sin", and yet VADER scores the sentence all but neutral, while most human readers would likely rate this one of the most negative in the data set.

Though not in the previous example, it is possible that some of these erroneous results may stem from the fact that the way VADER is being utilised in Sentiplot means that it has no context for any given sentence, they are all processed stand-alone and so VADER cannot build up any running themes or patterns to aid in analysis.

To conclude, this experiment raised potential issue with the usage of VADER, suggesting an alteration in Sentiplot's implementation but also questioned the overall effectiveness of VADER as a sentiment analysis tool given some rather unusual results upon close inspection. The next section partially opens up this discussion to a wider audience, allowing comment on the final results produced by VADER and some comments on its processes.

4.4 Reader Analysis & Reflection

This section covers the analysis of a range of Shakespeare plays by human readers (mostly identified to be Theatre and/or Literature students) and their personal comparison of the the Sentiplot output with their own expectations of what a graph of a given play should be. The goal here was to bring a human element to the reviewing of Sentiplot and VADER. A brief background of the study, its goals, and task was given to each person who answered any of the questions provided. Some asked further questions and subsequently commented on the techniques used for analysis and graph production.

4.4.1 Reader Analysis

Readers were first asked if they felt they knew each play well enough to attempt to plot a rough arc for each on a set of axes, with a focus on any major emotional events they could identify. Some participants opted to answer questions on only a subset of the plays, as they may not have read or otherwise cannot remember the other plays. In total, 22 result sets were gathered. This allows qualitative analysis to be carried out around how effectively VADER performed from a human perspective and to attempt to gather further comment from participants on the effectiveness of Sentiplot.

Each participant was given a blank set of axes (with the same scale as was used for VADER's results). They were asked to draw what they thought a curve of each play would look like, based upon any and all major events they can recall. They were then asked to label and/or explain any notable peaks and troughs in their curves. A small selection of the curves drawn are shown in figure 11, paired with the corresponding output form VADER. In general, all the curves drawn by participants were wildl different

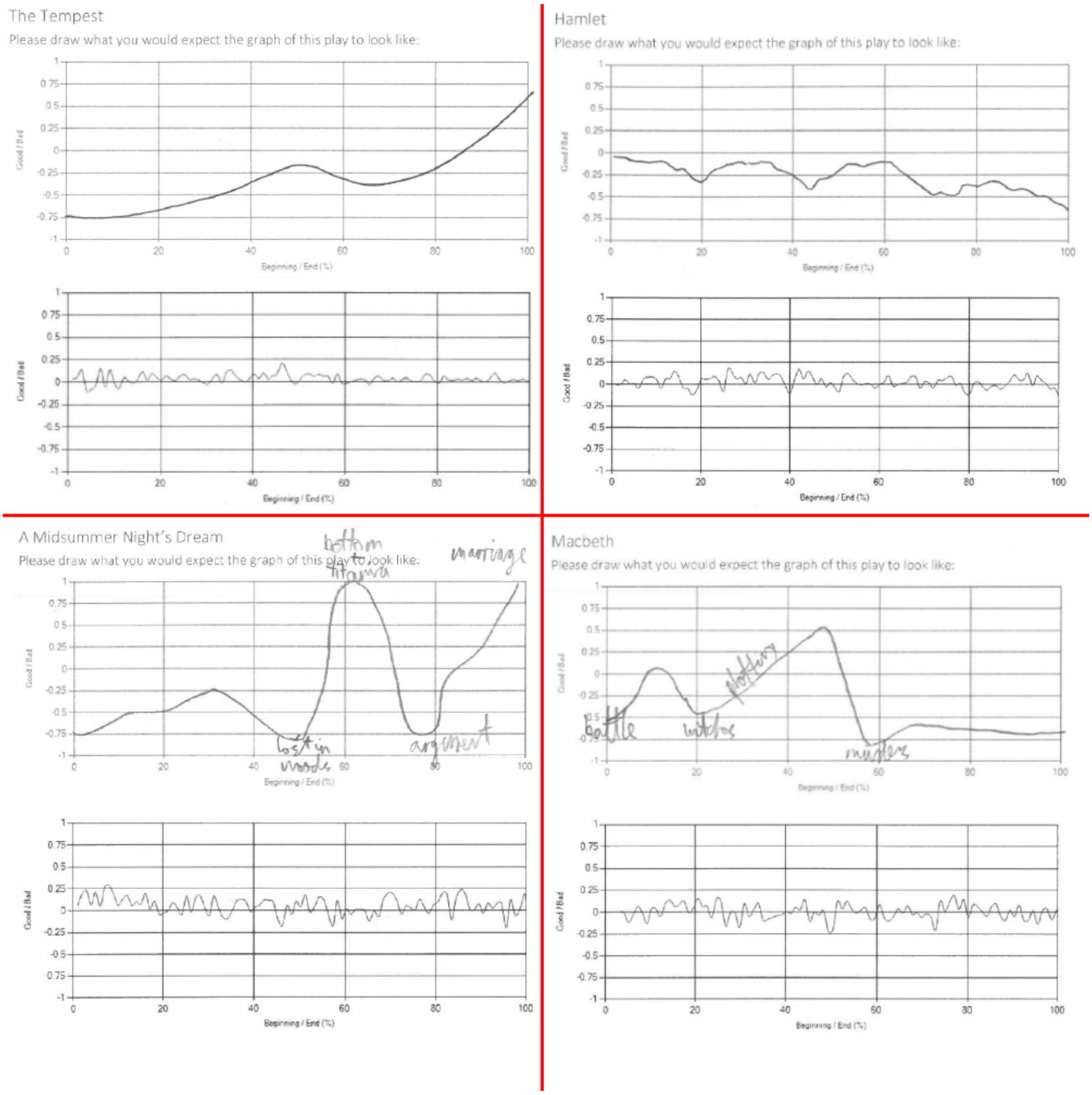


Figure 11: Readers (top of each pair) in general drew curves bearing little to no similarity to those produced by VADER (bottom in each pair)

to the VADER output for the same texts. All VADER's curves stayed mostly in the range of +/-0.25 on the Sentiment axis, whereas most participants drew curves using nearly the full extent of the y-axis. It was to be expected that participants would use the full axis, but it was not expected before setting out on this experiment that VADER's graphs would be quite so restricted. This is not to say that VADER doesn't register any sentences as reaching the more extreme values: when examining the per-sentence scores in the table view, it does, but these values get flattened when they are grouped together. On closer inspection, taking Macbeth as an example, approximately 30% of the sentences were outside the +/-0.25 range, however just over 60% of those *within* that range were zeros: registered by VADER as completely neutral sentences. When taking averages, these high volumes of zero-scored sentences reduced the impact on the output graph of higher scores greatly.

Even with this failure, attempting to look at relative differences along the two curve sets (those drawn, and those produced by VADER) still doesn't yield much similarity. The only immediately obvious correlation is, again, in Macbeth where both VADER and a participant show a sharp dip at around the halfway mark (labeled on the drawn curve as "murder"). This is nearly the only obvious match.

Kurt Vonnegut suggested in his lecture that Hamlet ought to have a somewhat flat curve. Sentiplot supports this at least visually, however, some of the drawn graphs show a very clear descent, logically coupling Hamlet's madness with negative sentiment. Sentiplot does in fact also show some extreme peaks and troughs, however these are momentary in the overall context of the script, and thus are averaged out and do not appear in the final output. Depending on the scope at which you examine, but Sentiplot has elements of agreement with both Vonnegut's flat prediction *and* participant's more curved graphs.

4.4.2 Reader Reflection

Participants were later asked to comment on the similarity (or dissimilarity) between their own graphs and those from VADER. There was a general consensus that VADER's graphs lacked the ability to show various details and intricacies of the literature or were otherwsie just plain wrong. One particpant noted the failure of the system pickup on tonal dissonances within a Midsummer Night's Dream - this is assumedly a reference to the multiple plot lines within the play and their sometimes conflicting emotional tones. If VADER had a greater capability to distinguish plot lines and characters within a text from one another, it's highly likely that each plot within a text would have its own curve.

It was also said that the Sentiplot graphs failed to pick up on even the extremities of character emotions (namely in Romeo and Juleit) and also in the overall tone of a play (Hamlet, showing as being relatively neutral when expected to be generally negative).

Another particpant mentioned that they felt undertones in the text were totally lost in the graphs (though this is perhaps to be expected) and that they failed to differentiation netween specific known periods of happy/sad narritive segements. One specific example of this is seen at the end of Romeo and Juliet - a decidedly negative section of literature, at least at surface level. Of the two responses for Romeo and Juliet received, both finish the play below zero, with one plummeting the sentiment score down to maximum negative. By contrast, VADER's graph actually *rises* at the end, the second highest point in the entire graph. This is a major error, of course, but after seeing the results, the text processed was examined closer and it was found that Sentiplot had failed to strip out the sizable Project Gutenberg pre- and postamble in the processed HTML file. It was the postamble (making up the final 5% of the text) that gave rise to this upward spike where there was otherwise a negative trend that met readers expectations. Unfortunately it was found that this was not an isolated issue and is covered in more detail in section 5.

A few participants noted that some of VADER's graph appeared to have little to no correlation with the text at all, with some attributing this to a computer program's lack of the ability to experience or empathise with literary art. It was also suggested that a theatrical script is likely to be a less than ideal format for attempting to produce a curve in this way, with one participant pointing out that scripts do not contain significant amounts of description - they only contain dialogue, with a small (albeit, varying) amount of description present in the stage directions; novels on the other hand innately contain description as it is the primary information is conveyed to the reader.

4.5 Early Modern Vs. Modern English

In this experiment, "varded" variants of six Shakespearean plays were processed by Sentiplot. The varded texts are modern English variants of Shakespeare's texts, the goal here being to compare just how much of an impact the difference in language (early-modern English vs. modern English) has on the ability of VADER to analyse sentiment.

VADER is not trained to work on the likes of formal literature. In fact, it is trained to work on highly modern and informal text, to such an extent that it can understand and analyse Internet slang and emoji. Due to this fact, this experiment was carried out to test if VADER was able perform equally or at all as well on a different form of English, that being the Early Modern English found in the Shakespeare plays processed for sections 4.4 and 4.6. The assumption is that VADER will perform better on modernised texts, as that is what its models have been trained to process.

The original Project Gutenberg texts of the plays gave graphs that did not hold much variation - processing the moderised text overall gave similar graphs at first glance, similar but with exaggerated peaks and troughs with the overall form is still held. However, finer inspection will show that in fact, there's a number of new features visible in most plays that were not present in the original text. For example: in the latter 40% of the original text of *The Tempest*, the graph has less varitation, staying practically in the range of zero to 0.1 - in direct opposition, the varded text has a number of sizable peaks and troughs, with a maximum magnitude score of approximately positive 0.3. Similarly, in *A Midsummer Nights Dream*, just after the halfway mark, there appears to be a fairly distinctive negative period for a while, a feature that isn't shown at all on the original text (in fact, it appears to go become more *positive*).

These new additions seen in most graphs demonstrate VADER's limited ability to process non-modern English text, causing it produce values erring much more toward neutral zero-values than it should. It is still reasonable to attribute the restricted nature of the these original graphs, at least partially, to the nature of the text, them being scripts, not novels, however it can still be seen that processing modern English is gives more meaningful results. This is evidenced in the huge variation (nearly full -1 to 1) of the graph of Cinderella in figure 7 which is also written in modern English.

4.6 Shakespeare: Comedies Vs. Tragedies

As mentioned in section 4.4, Shakespeare plays have been a key source of analysis text. These have generally yielded somewhat restricted graphs. The motivation for these tests came from the thought to force the ability to tell apart different plays by their genre. Namely to comedies *should* show generally more positive curves and similarly, tragedies *should* show more negative curves accordingly. To attempt to display this, three comedies (*Twelfth Night, The Tempest, A Midsummer Night's Dream*) and three tragedies (*Macbeth, Hamlet, Romeo & Juliet*) were processed to give their repsective graphs. There is some debate on the specific genre of *The Tempest*, it now often being referred to as a l̈ate romancë, a subsection of Shakespeare's comedies leaning more toward a ẗragicomedyïn genre. Nonetheless it ought to still produce a difference in curve to that of the three tragedies.

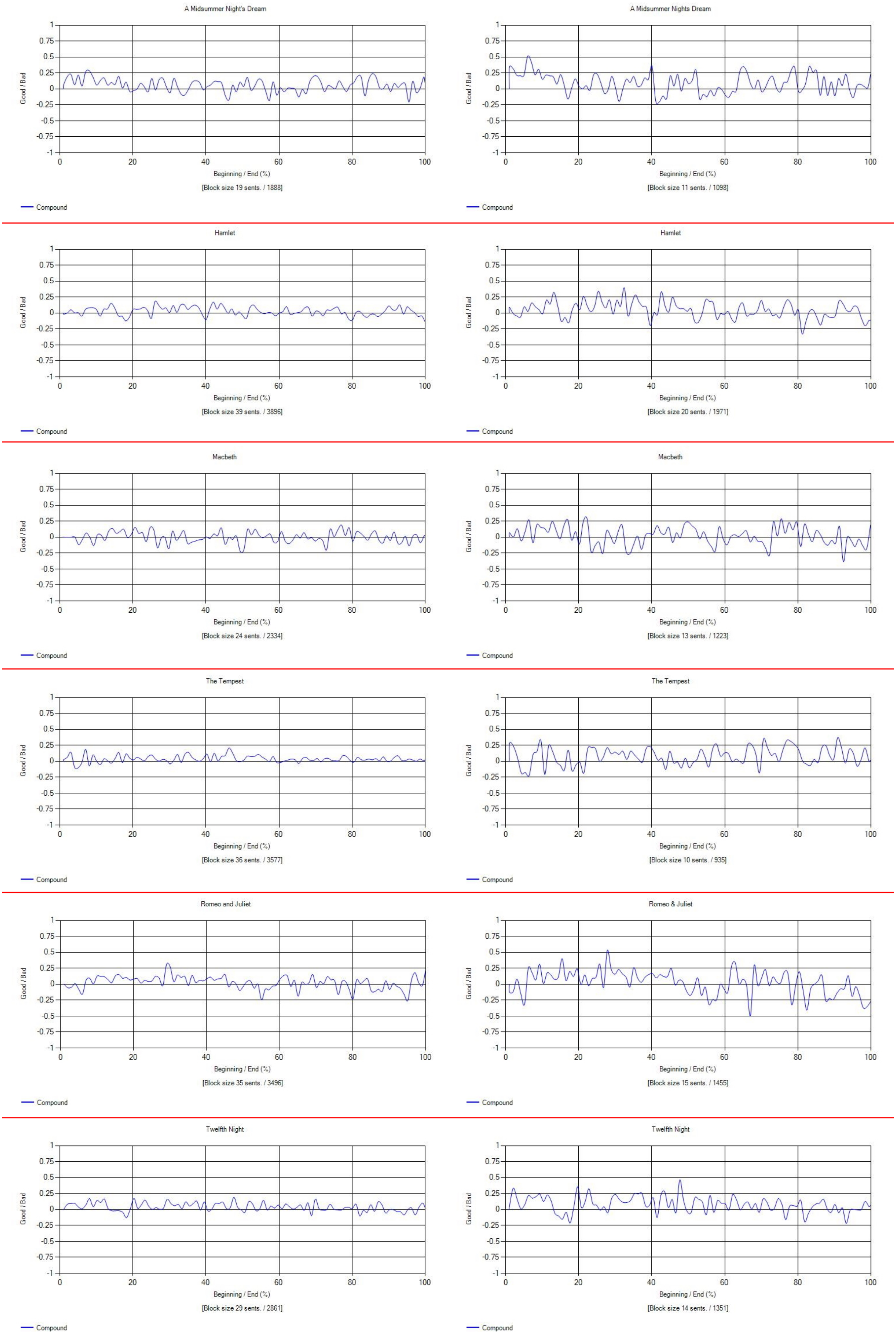


Figure 12: Original (early modern English) text graphed on the left, along shide hand-modernised text on the right for the six varded texts.

5 Project Gutenberg Formating

This section will examine the problems that were encountered using and processing corpora from Project Gutenberg. Chosen as the primary source of corpora, Project Gutenberg provides free, uncopyrighted literature for use in multiple formats. Early in development, Sentiplot had the capability to process plain text files only as this was deemed the simplest and easiest way to input text into the application. This was later found to be inadequet for the purposes of identifying and seperately processing individual chapters.

By chance, it was found that the HTML text provided by Project Gutenberg for both *Macbeth* and *Hamlet* (two of texts used to test the program during development) used HTML header tags to format Act and Scene numbers. It was also fonud that the preamble and postamble were enclosed in ¡pre¿...¡/pre¿ tags exclusively in this texts. An HTML file parser was then built in to Sentiplot to strip out unneeded information (pre/postamble, HTML tags, etc), decode it, and divide into chapter sections based upon these header tags. This worked for a number of texts, however, not all. It was taken for granted and assumed that Project Gutenberg used this as a standard format for all HTML file published - unfortunately this was not the case and sound of the graphs shown in this report are erroneous due to this.

Some of the texts lack any chapters that are able to be analysed and most have multiple chapters that either contain a majority of preamble/postamble or, are a single line of text or whitespace.

6 Conclusion

This is a conclusion.

References

[1] Shabaz Anwar. Sentiment analysis versus emotional analysis: Same or different?

[2] Ankit Rohatgi. Web plo digitizer tool.

[3] Kurt Vonnegut. Lecture to case western reserve university.