2013

Automatic Methods for Tracking Sentiment Dynamics in Plays

Eric Thomas Nalisnick
Lehigh University

AUTOMATIC METHODS FOR TRACKING SENTIMENT DYNAMICS IN PLAYS

by

Eric T. Nalisnick

A Thesis

Presented to the Graduate Committee

of Lehigh University

in Candidacy for the Degree of

Master of Science

in

Computer Science and Engineering

Lehigh University

May 2013

This thesis is accepted in partial

fulfillment of the requirements for the degree of

Master of Science.

____
(Date)

____
Henry S. Baird (Thesis Advisor)

____
Daniel P. Lopresti (Chair, CSE Department)

Acknowledgments

I would like to thank my advisor, Henry Baird, for his generous support, guidance, and encouragement not only in regards to this project but my academic career in general. I would also like to thank my family for their enduring support.

Contents

List of Figures

Abstract

An investigation into algorithms for generalized, dynamic sentiment tracking in natural language is reported. Little progress has been made in the automatic analysis of literary fiction. However, accurate recognition and robust modeling of text's emotional content would allow for large scale analysis of the ever increasing number of digitized books in addition to other applications such as sorting, searching, and making book recommendations. But the challenge of human-like reading of these books not only includes computational hurdles but inherent ambiguity. Humans often argue over the correct interpretation of a work of fiction due to subjectivity. To attack this open problem, we resort to shallow statistical methods. We propose a method for tracking the sentiment of the interpersonal relationships of literary characters by leveraging text structure to estimate the direction of sentiment flow. Given structured dialogue (as found in a play), we know who is speaking and can perform standard sentiment analysis on their speech via an emotion lexicon. Furthermore, we can guess at who is listening and direct the sentiment accordingly, producing lists of a character's enemies and allies. Experiments on all of Shakespeare's plays are presented along with discussion of how these methods can be extended to unstructured texts.

Chapter 1

Introduction

Emotion is the defining quality of literature. A quality piece of literary fiction will drag the reader on a visit to emotional highs and lows. In the following work, we describe our attempts to computationally capture and model–with little to no human intervention–this 'roller coaster' of emotion. Modeling and understanding the emotional content and relationships described in natural language text remains a daunting technical challenge that is nearly as hard as generalized machine reading. Humans bring deep contextual knowledge to everything that they read. Although machines can quickly access huge databases that could store some, but probably not all, of this contextual data, these remain too incomplete and unstructured to match the inferences that humans routinely make [PnH10], leaving robust semantic extraction possible for only short passages [Smi03]. Since deep reading is not feasible without considerable human intervention and loss of generalization, shallow statistical analysis seems to be the only scalable alternative.

Statistical methods have been used for decades to model and analyze syntax. Automatic part of speech (POS) tagging and syntactic rule checking has been done successfully for English as well as many other languages [TM00]. The emotional analog of POS tagging is *sentiment analysis* via emotion lexicons, a method of shallow analysis that maps a given word to a *valence* value, a real number intended to capture the degree of emotional positivity or negativity usually associated with the word. For example, the WordNet Affect Lexicon (WAL) associates words with varying degrees of the six Ekman emotions: joy, sadness, anger, fear, disgust, and surprise [SV04].

Despite the widespread use of sentiment analysis in commercial applications, such as analyzing customer reviews, little work has been done on applying sentiment analysis to fiction, which is not too surprising. Insightful analysis of literature often challenges trained human readers let alone machines. In fact, automatic analysis of literature has, so far, been constrained to applications that either do not consider the content of the story (*e.g.* attempting merely to attribute authorship [ZZ07]), or that require the intervention of human experts (*e.g.* making inferences based on word occurrence statistics [Rom94]). There appears to have been little progress in robust, automatic methods to extract and make use of plot, setting, and relationships among characters. We hope that progress might be possible by decomposing a text into paired character interactions and then analyzing the sentiments expressed in dialogue.

Chapter 2

Sentiment Analysis

Sentiment Analysis (SA) (or *Opinion Mining*) [PL08] is a natural language processing (NLP) technique that aims to extract subjective information–usually the emotion or feeling–from raw text . It is most commonly used to investigate the general emotional context of a document or, if a single writer is the subject of interest, to judge how the author feels about the topic of the text. This latter use has lead to much work in commercial applications. SA has been able to successfully classify customer opinions contained in product reviews [TNKS09] and social-media messages [PGMJ11]. Yet note by 'successfully,' we mean that SA techniques have been able to achieve the same accuracy as humans, approximately 70%, in classification tasks [PLV02]. This far from perfect level of success means that there is inherent ambiguity in SA. There is no gold standard as to what emotions a reader should be extracting from a text.

The best approximation of ground truth-ed sentiment content is user ratings, which usually accompany text in online reviews [PLV02]. For example, *Amazon* asks users to rate their purchases on a scale of zero to five *stars* in addition to providing a paragraph or more describing their experience with the product. Making the assumption that the opinions in the text are essentially labeled by the rating allows statistical machine learning techniques to be used for SA. In some of the earliest work in SA, B. Pang et al. used Naive Bayes, Maximum Entropy, and Support Vector Machines on bag of words features to classify movie reviews as positive or negative, achieving as high as 82.9% accuracy [PLV02].

Yet–unfortunately–most text does not come labeled, making traditional probabilistic classification impossible. In turn we must resort to knowlege-based methods: sentiment lexicons or lists that map words to a sentiment score. These lists are usually created via crowd sourcing. Participants are shown words and told to determine their polarity (i.e. positive emotion = +1, negative emotion = -1) and possibly to assign the word a valence score that represents degrees of emotional intensity (i.e. -10 = *most* negative, 0 = emotionally neutral, +10 = *most* positive) [KH04]. The responses from participants are averaged to determine the final sentiment scores.

There are a few other SA methods that have merit but are less widely used. For example, some SA algorithms are driven by recognizing syntax commonly associated with expressions of dis/pleasure [TNKS09]. Others make inferences via link analysis (*e.g.* one's views may be influenced by one's friends') for sample labeling or to track the speed at which sentiment moves through networks [KNS+08]. Some refine sentiment into categories associated with specific emotions [SV04], and others try to distinguish the source of the sentiment or object to which the sentiment is directed [TM08]: *e.g.* when a user writes, "we love the interface but hate the customer support," the two cases need to be teased apart to attribute dis/pleasure accordingly. There is also interest in detecting fraudulent reviews by identifying disingenuous exhibitions of sentiment [MPNdL10].

Our work will not use these latter methods since they usually are domain dependent, and we are focusing on highly generalizable techniques.

2.1 Computational Analysis of Literature

Turning our attention now to the automatic analysis of literary fiction, we find few clear successes. Perhaps the most intensively studied application has been authorship attribution (AA). Typically, after deleting "stop words"[1], an AA system tags each remaining word with a POS and groups these tags into N-grams [ZZ07]. Machine learning methods model the N-gram distributions separately for each author of interest and the resulting classifiers attempt to validate authorship claims and suggest authors for anonymous works. AA systems have, famously, been wielded in debates over Shakespeare's true identity [ZZ07].

To the best of our knowledge, little has been accomplished in automatic analysis of the content (*i.e.* plot, setting, characters, etc.) of works of fiction. Mutton has made progress in understanding social structures described in fiction by adapting methods for extracting social networks from Internet Relay Chat (IRC) to mine plays for their networks [Mut04]. Novels would seem far harder, but Elson et al. developed a reliable method for speech attribution in unstructured texts, which they later employed to extract social networks from Victorian literature [EM10][EDM10][ACJR12]. This line of work, we believe, is valuable because fiction, essentially, is a description of character interactions, which can be modeled by Mutton's and Elson's methods.

Likewise, there has been little work in analyzing the emotional content of fiction.

[1] Oversaturating words whose use is motivated more by proper syntax than by meaning: *e.g.* 'the', 'a', 'to'.

Mohammad tracked emotions over time in several genres of English fiction including fairy tales and novels [Moh11] and argues for SA's worth in search and summarization of literary texts. We now extend his work by investigating the analysis of character-to-character interactions over time in plays.

2.2 Full-Text Analysis of Shakespeare's Plays

Our primary goal has been to explore what sentiment analysis, applied fully automatically—with a minimum of hand-crafted adaptations—can reveal about the narrative content of fiction. Since there is no definitive ground truth as to what emotions a story should elicit (as mentioned previously), we could not immediately apply machine learning based SA. Crowd-sourcing possibly could have been used to produce labels that denoted what emotions the *average* reader usually feels when reading some play, but due to time constraints, we will leave this area to future work.

The lack of labels forced us to use a knowlege-based SA approach. We chose the *AFINN* word list for sentiment analysis [Nie11], which contains 2477 English words, labeling each with a *valence* value, an integer between -5 and +5 (for example, *catastrophic* has a value of -4, *outstanding* has a value of +5.), because the list provides valences (not just polarities) and is designed for social media analysis. We did not modify this modern word-list before applying it to 16th C. literature in the hope that our methods might work at least as well on any play written since.

Our first experiment was a simple 'sanity check' on the worth of SA's application to literature. By using the genre classes (which are relatively well-agreed upon by Shakespearean scholars) as labels, we tested if sentiment could differentiate comedies from tragedies. For each play, we calculated the average word valence (AWV) by summing the valence values for all words in the play and then dividing by the number of the play's words in AFINN. Figure 2.1 shows that comedies generally have a higher AWV than tragedies (as we expected). The blue dotted line marks the average value for comedies, 1.55, which is almost double the average AWV for tragedies, 0.84, as marked with the red dotted line. If we created a threshold classifier by setting the class boundary to an AWV of 1.0, the results show that the classifier would be correct in 20 out of 24 instances (83% accuracy).

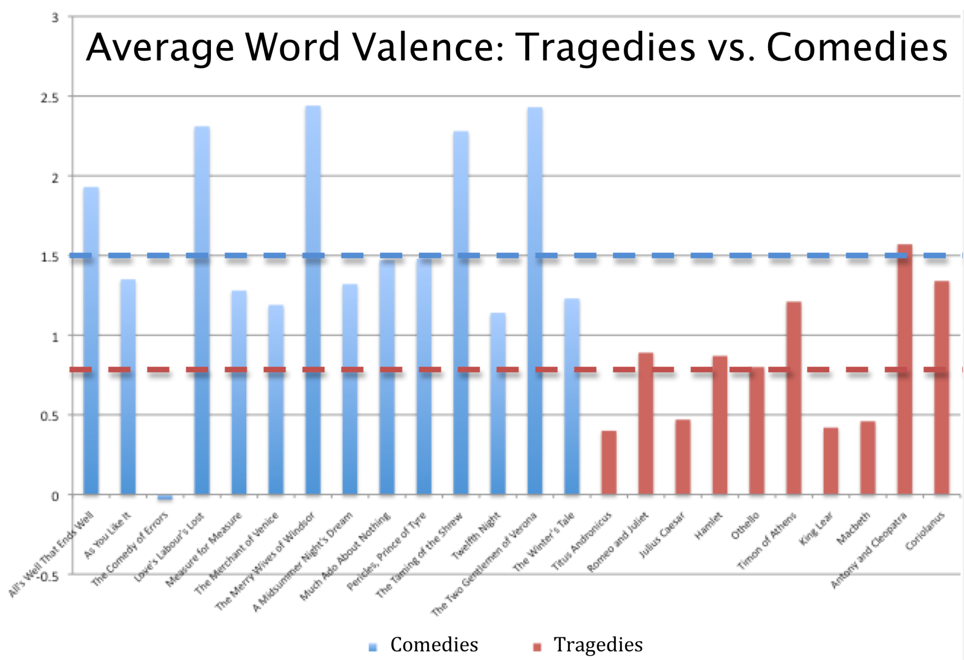

Figure 2.1: For each play, the valence of each word was summed and then divided by the number of words in both the word list (AFINN) and the respective play. The result is the average word valence (AWV), which seems to discriminate well between tragedies and comedies. The blue line marks the average AWV for comedies and the red line represents the same value for tragedies.

There are several notable observations in the results. *Titus Andronicus* has the lowest

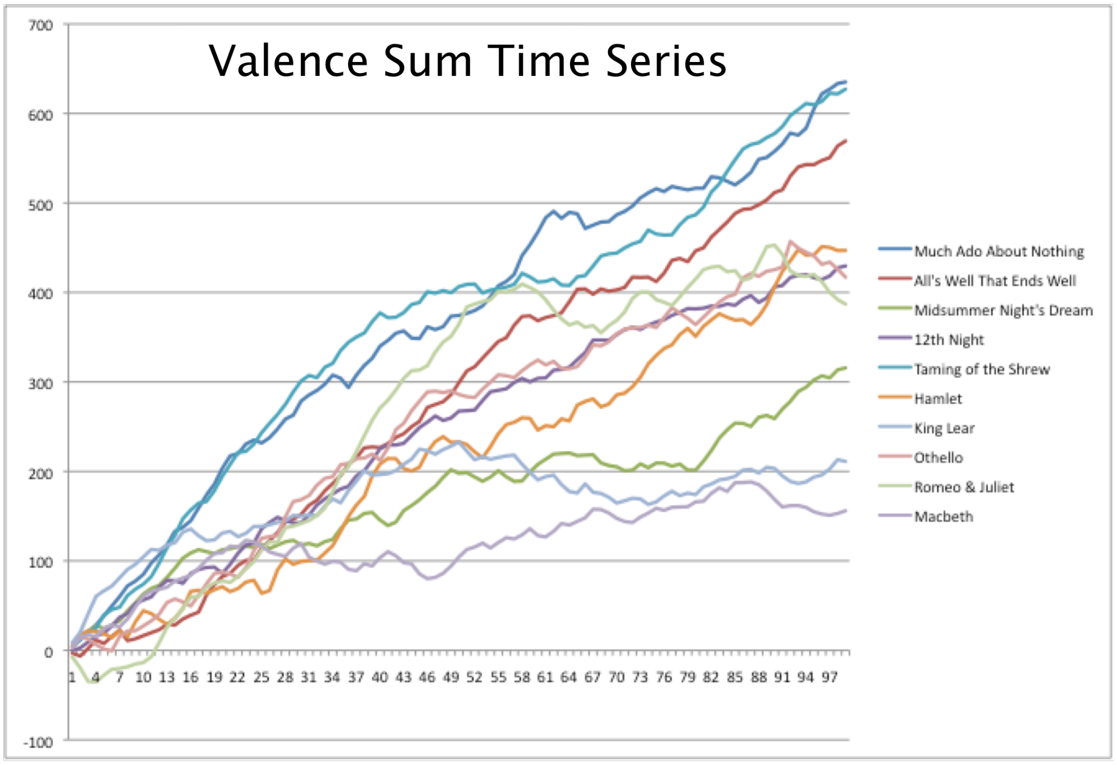

Figure 2.2: For each play, the valence value of each word was summed over time and was then normalized by breaking each play into percentiles. Tragedies tend toward the horizontal while comedies have a consistently positive slope.

AWV of the tragedies (0.40), which is a success since the play is widely considered to be Shakespeare's bloodiest and most violent work. But also notice that *The Comedy of Errors* (COE) has an even lower AWV of -0.03. There are several possible explanations for this outlying value. The play was one of Shakespeare's earliest works (1st to 3rd) and therefore might be stylistically immature. Furthermore, upon examining the play's content, the plot features farcical but dark elements such as beatings, infidelity, theft, and madness, which probably have been taken at 'face value' by the shallow analysis. Yet there are some critics that might say COE's AWV value is not an outlier. Gwyn Williams claims, in her essay *The Comedy of Errors Rescued from Tragedy*, that "A careful analysis of this play, however, shows that it might easily have worked out as a tragedy" [Wil64]. The sentiment analysis seems to support her claim.

Given that sentiment analysis passed our initial test, we wanted next to examine how the sum of the valence values (accumulated from the start of the play) change over time, and test whether fluctuations in this corresponded appropriately with events in the respective play. Figure 2.2 shows the valence-sum time series normalized for the length of the play (by breaking the play into percentiles and averaging each time step within the one hundred intervals). Some plot features are conspicuous. For instance, the dip at the beginning of *Romeo & Juliet* corresponds to the opening fight scene. Also notice sharp descents at the ends of both *Othello* and *Romeo & Juliet*, corresponding to their sudden tragic endings. Overall, we can see that most of the tragedies have settled to the bottom, taking on a nearly horizontal trajectory as the play progresses (*e.g. Macbeth*, *King Lear*).

Chapter 3

Social Networks

A social network is a graph $\mathcal{G}$ that models the social structure of a community by encoding connections between actors, which could be individuals or groups depending on the domain. The actors are represented by the set of *nodes* (also called 'vertices') $\mathcal{V}$, and the connections are represented by the set of *edges* $\mathcal{E}$ such that $e_{i,j} \in \mathcal{E}$ denotes a connection between nodes i and j. The social networks described are *undirected*, meaning $e_{i,j}$ implies $e_{j,i}$. Yet note that this implication does not hold for *directed* networks. Economic behavior, criminal activity, disease transmission, and social phenomenon (among other applications) have been accurately modeled by social networks [JA06].

In the domain of literary texts, Paul Mutton was the first to demonstrate social networks could easily be extracted from structured texts but did little analysis of his results [Mut04]. Elson et al. significantly improved upon the line of work by demonstrating character networks could be mined from unstructured texts [EDM10]. The method was used to extract networks from sixty Victorian novels. Analysis was then performed on these networks to show counter-evidence to some popular critical beliefs about novels from this time period.

S. Gil et al. extracted networks from 190 plays in Project Gutenberg and 951 movie scripts from The Internet Movie Script Database [GKS]. They then calculated structural properties of the networks such as number of nodes, single character centralities, and entropy of node degrees among others. Several classification tasks were attempted using this data. Experiments such as distinguishing between movies with good vs. bad critic ratings were unsuccessful (46.8% accuracy). However, distinguishing movies from plays was performed with 89.2% accuracy and distinguishing the plays of George Bernard Shaw from the plays of Shakespeare was done with 100% accuracy. Gil et al. also tried genre classification and observed varying results. Horror movies were identified with the highest accuracy (82.5%).

3.1 Shakespearean Social Networks

To construct a social network from text, who is speaking and to whom he or she is speaking must be known for every instance of dialogue. This is hard to do with unstructured texts, such as novels [EDM10]. However, the task is considerably easier in the case of plays because they are structured: every line is tagged with its speaker. Jon Bosak has converted thirty-seven[1] Shakespearean plays to Extensible Markup Language (XML)[2], making the processing of these texts especially easy.

[1] Shakespeare wrote thirty-eight plays. *The Two Noble Kinsmen*, a rare play, is missing from Bosak's data set.

[2] http://www.ibiblio.org/xml/examples/shakespeare/

3.1.1 Construction Method

The first automatic social network extraction from literature was done by P. Mutton [Mut04]. He modified a program built to extract networks from Internet Relay Chat (IRC) to do the same with plays and demonstrated it on the works of Shakespeare. His program had a strict definition of social connection: for two nodes (i.e. characters) to be connected, the two characters must have adjacent speaking parts at some point in the play. This requirement assumes that adjacent lines imply that the two characters are speaking to one another, which seems like a reasonable assumption. However, inspecting *The Tragedy of Hamlet*'s social network when it is generated by this technique reveals some obvious flaws. For example. the characters Rosencrantz and Guildenstern are inseparable throughout the play so much so that their personalities and character traits are indistinguishable. In fact, there is never a need for the two to talk to one another directly. And since the two never speak, the strict definition of connectedness never is met, and thus there is no edge in the network between Rosencrantz and Guildenstern even though the two are obviously close friends.

A better method is to track the line distance between speakers and use the distance to calculate a *closeness* measure for two characters [GKS]. A window of the ten previous lines was examined after every instance of speech, and for every pair of characters an interaction score is kept. Every time two characters speak within ten lines of one another, the inverse of their line distance is added to their interaction score. For example, one ($\frac{1}{1}$) is added to the score when characters speak adjacent lines, and if their lines occur at the maximum distance, ten, 0.1 ($\frac{1}{10}$) is added to the score. Notice that the closeness extraction method does not process one word of dialogue. The play's sequence of speaker tags is the only required input, making the network mining process very fast. In addition, this technique solves the Rosencrantz and Guildenstern problem by detecting that the two have lines in close proximity and therefore are socially connected.

Moretti's handcrafted *Hamlet* social network [Mor] was used to test the accuracy of the network produced by the above method. Three edges were found in the above method's network that did not exist according to Moretti's network, and this discrepancy was rectified by creating a threshold value of .45. If the interaction score of two characters does not exceed this threshold, no edge exists between them.

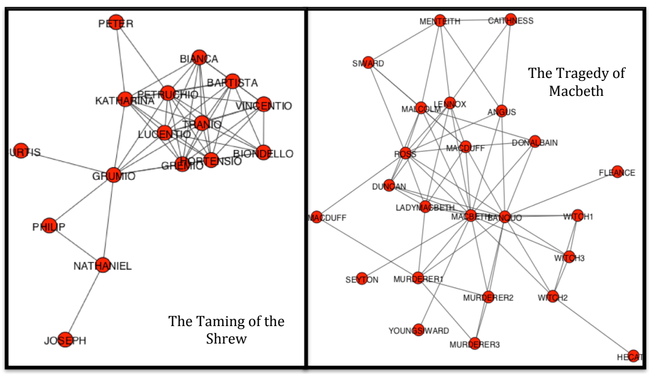

Figure 3.1: The social networks for *The Taming of the Shrew* and *The Tragedy of Macbeth* are shown above. The comedy (left) has a a lower number of characters and a higher connectivity than the tragedy (right). The structural features of each network are as follows. *Taming of the Shrew*: vertex count = 19, edge count = 74, avg. path length = 1.78, density = 0.43, diameter = 4, max clique size = 10, PageRank of main character = .08. *Macbeth*: vertex count = 24, edge count = 78, avg. path length = 1.82, density = 0.28, diameter = 3, max clique size = 6, PageRank of main character = 0.12.

3.1.2 Structural Properties

Once the thirty-seven networks were extracted, the next step was to calculate various structural features in the hopes that some of them would differentiate the three genres.

Global Features

Simple, macro-features of the network were obtained first: node count and edge count. Node count represents the number of characters in the play. We believed this feature would be important because histories, and sometimes tragedies, have a large number of characters because the plot often involves an entire royal court, pulling in everyone from family members to servants. On the other hand, comedies tend to have a smaller number of characters because comedy draws its effect from the constant interaction of a few diverse individuals. Therefore we hypothesized that histories will have a large node count, comedies will have a small node count, and tragedies will be somewhere between the two other genres.

Edge count does not have strong motivation for its inclusion besides the fact that edges are the other fundamental components of graphs. Edge count does contribute to the connectivity of a graph, a very important feature that will be discussed in the next section, by measuring the total number of connections in the graph, but we believed network density and average path length would represent the connectivity better than edge count and be more useful during classification.

Connectivity and Cliques

The *connectivity* of a social network is a measure of how easily one node can be reached from another [JA06]. Both robustness of and distance within the graph must be considered. Two nodes that are connected by numerous short paths have the highest connectivity, and nodes that are connected by only a few long paths have the lowest connectivity (or highest disconnectivity). We hypothesized that at least one of the three measures of connectivity would be the best inter-genre discriminant: *diameter* $W(\mathcal{G})$, *density* $D(\mathcal{G})$, and *average path length* $APL(\mathcal{G})$. All three statistics were calculated for each network as they are defined as follows. Let $d(v_i, v_j)$, where $v_i, v_j \in \mathcal{V}$, be the shortest distance between v_i and v_j ($d(v_i, v_j)$=0 if v_i and v_j are not connected). For a network $\mathcal{G}(\mathcal{V}, \mathcal{E})$,

$$W(\mathcal{G}) = \max_{i,j}(d(v_i, v_j)),$$

$$D(\mathcal{G}) = \frac{|\mathcal{E}|}{\binom{|\mathcal{V}|}{2}},$$

$$APL(\mathcal{G}) = \frac{1}{|\mathcal{V}|(|\mathcal{V}| - 1)} \cdot \sum_{i,j} d(v_i, v_j).$$

We chose max clique size as the fourth statistic for inferring connectivity. A *clique* is a complete subgraph (i.e. a subset of vertices such that every vertex is connected to every other vertex in the subset), and hence the max clique size is the number of nodes in the largest complete subgraph. The clique decision problem is NP-Complete, but the relatively small number of nodes in the graphs (less than forty) makes the computation tractable for these experiments.

Our intuition on the value of measuring connectivity had its origins in the nature of tragedy. Tragic drama stems from a lack of connectivity. The network is divided into factions and each faction has a plot against the other, limiting interaction. For example in *The Tragedy of Macbeth*, Lady Macbeth and her husband have a secret plot to kill King Duncan. They keep this information between themselves (for obvious reasons) and hire henchmen to help them in their takeover. Then when the murder occurs, the lords split into opposing groups: Macbeth versus Macduff (and their respective followers). we believed this division of ”good” vs ”evil” would be conspicuous by way of lower densities, longer average path lengths, longer network diameters, and a small max clique size (with respect to the number of characters).

Histories, we hypothesized, would have similar connectivity to tragedies. There are often many characters involved in the plot, making it infeasible for the play to have the same high degree of connectivity as comedies. Furthermore, histories often feature hidden agendas and rouge members of the court, the same dramatic elements that socially isolate characters and lead to less connectivity in the graph.

Conversely, comedy is a product of over-connectivity. In Shakespeare's comedies, there is often a small cast of characters with very diverse personalities and traits, and these characters are forced to interact, resulting in humorous consequences. For instance, *The Taming of the Shrew* describes Petruchio's, a high-spirited, overly enthusiastic man's, mission to seduce Katherina, an ill-tempered and sharp-tongued woman. Humor ensues because of the continual–and almost implausible–interaction of these two very different people. This state is represented in the network by an abundance of activity and connections that probably are not common in the tragedies. Thus, we expected to see higher densities, shorter average path lengths, shorter diameters, and a large max clique size (considering the number of characters).

One concern we had with these hypotheses and intuitions laid in the indistinctness of comedy's definition. The only guarantee with comedies is that there will be very few deaths, if any, in the plot. A large portion of the comedies are farcical and devoid of death or even the threat of it. However some comedies describe sorrowful situations that could easily be revisioned as tragedies. For instance in *A Winter's Tale* and *Much Ado About Nothing*, sorrowful endings are avoided by the revelation that the leading female character merely faked her death, allowing her to return at the climax for a cheerful ending.

Main Character and PageRank

Google's seminal *PageRank* algorithm calculates a node's popularity by taking into account the number of edges that include that node. Or in other words, PageRank calculates the probability of reaching a node at any given time while randomly walking through the network [PBMW99].

The intuition behind the inclusion of this statistic is that the main characters in comedies will have lower PageRanks than the main characters in histories and tragedies due to the fact that comedies create affect through interaction whereas histories and tragedies are portraits of one large personality. For example, *A Midsummer Night's Dream* is entertaining because of the discord between three sets of couples. If one of the couples is removed, the story has a blatant gap but does not completely collapse.

In tragedies and histories, however, there is often a central character who motivates the drama by himself. *Othello*, *King Lear*, *Macbeth*, and *Richard III* all are the primary source of momentum in their respective plays. The tragic element usually is a result of some ill-choice on the part of the namesake character: Macbeth allows himself to become consumed by power, Othello lets jealousy prevail over reason, and King Lear chooses pride instead of humility. Of course there are supporting characters influencing these decisions, but the general consensus is that the main character is not powerless. They are the pivot, the axis on which the drama turns. Thus, we expected them to have a higher PageRank than the main character of a comedy[3].

[3]The character who speaks the most lines was chosen as the main character for plays without a clear central figure i.e. Romeo was chosen as the main character for *Romeo and Juliet* because he has forty-five more speeches than Juliet.

Chapter 4

Predicting Genre via Network Analysis

4.1 Three-Genre Classification

The first series of experiments tested if any supervised learning techniques, when given the aforementioned network attributes for each of Shakespeare's thirty-seven plays (10 tragedies, 10 histories, and 17 comedies), could learn to classify each into its proper genre[1]. First, several attribute analysis techniques were performed to evaluate the hypotheses put forward in section three. Then supervised learning was performed using all features and then for various subsets of the features based on the attribute selection results. The Weka tool package was used to perform all data analysis[2].

[1] Genre classification from NoSweatShakespeare.com
[2] http://www.cs.waikato.ac.nz/ml/weka/

4.1.1 Attribute Selection

To test the claims made in section three, a variety of attribute evaluation and selection techniques were run. The first, *Correlation Feature Selection* (CFS), searches the attributes for a subset of uncorrelated features that are highly correlated with the class labels. The second, *Principal Component Analysis* (PCA), analyzes the set of data points for the Eigen vectors of the dimensions with the greatest variance. The last, *Information Gain Ratio*, calculates the information encoded by the label and normalizes for the information encoded by the feature. See Figure 4.1 for the results. All three methods agreed

Method	Attribute Ranking
CFS Subset Eval	(1) Vertex Count, (2) Density
PCA	(1) Vertex Count, (2) Avg. Path Length, (3) Density
Info. Gain Ratio	(1) Vertex Count, (2) Density, (3) Avg. Path Length

Figure 4.1: Three attribute selection techniques were performed on all seven attributes. Only the top three attributes are shown. *Vertex count* was selected as the most useful feature.

that vertex count allows for the best inter-genre discrimination. PCA calculated it accounts for 55% of the variance. Density was chosen as the next best attribute and average path length as third best (accounting for a combined 29% of the variance). It is reasonable that CFS did not output any of the other features relating to connectivity (diameter, average path length, max clique size) since the method discards correlated features.

We were curious about the discriminating power of PageRank so Weka's visualization feature was used to explore this feature manually. The feature space is show in Figure 4.2. PCA returned that PageRank accounts for 4% of the variance. Inspecting the distribution reveals that all three genres have much overlap in this attribute, and our hypothesis (the main characters of comedies will have the lowest PageRanks) is not supported. Instead, histories have the lowest PageRanks, which makes sense given their typically large cast.

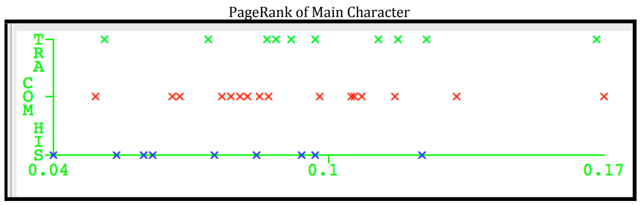

Figure 4.2: The figure above provides a visualization of the PageRank attribute. The green points are tragedies, the red points are comedies, and the blue points are histories. The feature space shows little potential for discrimination between genres (standard deviation = 0.03).

4.1.2 Supervised Learning

After evaluating the attributes, experiments were run to test how well genres could be classified by these structural features. To ensure we did not overlook some classification method that could be useful, six types of classifiers were trained and evaluated using *ten-fold cross validation*, i.e. dividing the training set into ten parts, using nine for training and one for testing, and averaging the classification accuracy over ten trials.

The number of input features to a supervised learning algorithm can greatly affect the accuracy of the classifier. Therefore, each classifier was run with seven, three, two and one attribute(s). The three attribute subsets were chosen by the rankings calculated in the attribute selection phase (section 4.1.1).

Three-Class Trials				
	Number of Attributes			
	7	3	2	1
J48 (Decision Tree)	64.86	67.57	67.57	67.57
JRIP (Rule Learner)	67.57	62.16	64.86	64.86
Clustering	56.76	51.35	43.24	59.46
Regression	62.16	67.57	67.57	67.57
Bayesian Network	62.16	62.16	62.16	67.57
Multilayer Perceptron	59.46	67.57	64.86	70.27

Figure 4.3: The accuracy rate of the six supervised learning methods with ten fold cross validation is presented above. The *multilayer perceptron* trained with one attribute, vertex count, was best at classification with a success rate of 70.27%.

The first classifier, *J48*, is a decision tree. The information gain heuristic is used to pick the attribute to split on at each depth. Pruning and subtree raising are used to combat over-training. The decision tree performed best with three or less attributes, achieving a relatively admirable 67.57% classification success rate in each of these trials.

JRIP, a propositional rule learner, was used as the second classifier tested. JRIP creates rules by undergoing a series of growing and pruning operations until entering an optimization phase when an error rate or a description length threshold is met. JRIP's accuracy was close to J48's, hovering between 62% and 67%, an observation that is not surprising since rule learners and decision trees usually perform comparably. However, unlike the decision tree, JRIP performed best using all seven attributes even though it created only two rules from them: If *vertext-count* $\geq$ 22, then *class* = *HISTORY*. Else, *class* = *COMEDY*.

Classification via clustering was the third method tested. Specifically, simple *K-Means* (finding the closest cluster centroid to an unclassified instance) with K equal to three and Euclidean distance was used. This method performed the worst, hovering around 50% no matter how many attributes were used during training.

Classification via regression performed surprising well overall. The Weka implementation uses *M5 pruned model trees*. This method creates a tree for every class, and the leaves of these trees are mapped to linear functions. For instance, the tree for the history class splits on one attribute, vertex count, sending instances with less than 21.5 nodes to linear model #1 (LM1), which is defined as 0.0151 $\cdot$ *(vertex count)* - 0.2143, and the others to LM2, defined as 0.0197 $\cdot$ *(vertex count)* + 0.064. Every instance is sent down each class tree, and the instance is assigned to the class whose function evaluates to the highest value. Regression performed slightly better by reducing the number of attributes but the reduction improved the accuracy by only about 5%. Regression is usually very tolerant of noise, which explains the even performance.

A probabilistic classifier was chosen fifth. Most flavors of probabilistic classifiers hinge on independence among attributes, but since diameter, density, average path length, and max clique size all correlate with connectivity (to varying degrees), a *Bayesian Network* classifier was the best choice. A key aspect of Bayesian Networks is their dependency network, a directed graph that models dependencies among features, which allows the method to recognize and cope with a lack of independence. This network structure can be learned through a variety of algorithms that leverage local score metrics, global score metrics, and conditional independence tests to search through possible structures. The Weka implementation used the hill-climbing method K2, a local score metric.

Yet despite our precautions to compensate for independence, the Bayesian Network learned a structure that assumed independence among attributes: a tree structure with seven leaves. The class attribute was at the root of the tree, and the other seven attributes were the leaves. No further edges existed to represent dependencies between any of the features used to gauge connectivity, an unexpected occurrence.

The Bayesian Network performed close to uniformly across each trial as expected (a Naive Bayes method would have performed precisely uniformly since the error rate is guaranteed not to increase with added attributes). The classifier leveled at 62.16% accuracy for the first three trials and then jumped to 67.57% for the one-attribute trial. Yet note that Bayesian Classifiers are supposed to improve with the addition of features because more information usually means a better understanding of the dependency structure of the attributes. We suppose this trend was not observed in our experiment because the Bayesian Classifier found no dependencies between any of the seven attributes, resulting in the aforementioned tree structure.

A *multilayer perceptron* was used during the final trial. These classifiers consist of a layer of input nodes, zero or more layers of hidden nodes, and a layer of output nodes. Each layer is completely connected to the preceding layer by directed edges. During training, the weights for each node's nonlinear activation function are learned through a method called back-propagation. Multilayer perceptrons are essentially a combination of linear classifiers that results in nonlinear discriminatory power. The best result, 70.27%, was achieved with two input nodes that were both fed with the vertex count of the instance. These nodes were then connected to three output nodes, one for each genre.

4.2 Comedies vs Tragedies

After running trials to attempt classification of all three genres at once, pushing the accuracy rate by limiting the data set to two classes was attempted. The first experiment left out histories, resulting in a data set of ten tragedies and seventeen comedies. As with three genres, attribute selection was first performed (with the same three methods) to further understanding of what are the most powerful features. Then the same classifiers were trained and evaluated again using ten fold cross validation. It was hypothesized that tragedies vs. comedies would be the easiest to distinguish when the problem is reduced to two classes.

4.2.1 Attribute Selection

CFS, PCA, and information gain were run on the set of seven attributes. Figure 4.4 shows the top three ranking attributes given by each method (if more than three were given). Average path length was chosen as the best discriminant by two out of the three methods

(PCA and Info. Gain). While we found this result surprising at first, further investigation

Method	Attribute Ranking
CFS Subset Eval	(1) Vertex Count
PCA	(1) Avg. Path Length, (2) Density, (3) Vertex Count
Info. Gain Ratio	(1) Avg. Path Length, (2) Vertex Count, (3) Edge Count

Figure 4.4: Three attribute selection techniques performed on all seven attributes. Only the top three attributes are shown. *Average path length* was selected as the most useful feature.

via visualization showed that there are some tragedies with a low character count, such as *Othello*, that complicate the decision boundary[3]. With average path length being slightly better than vertex count, the hypothesis that information does not travel as easily through tragedies as it does comedies, a product of secret plots and factions, is supported. Figure 4.5 shows side-by-side visualizations of vertex count and average path length.

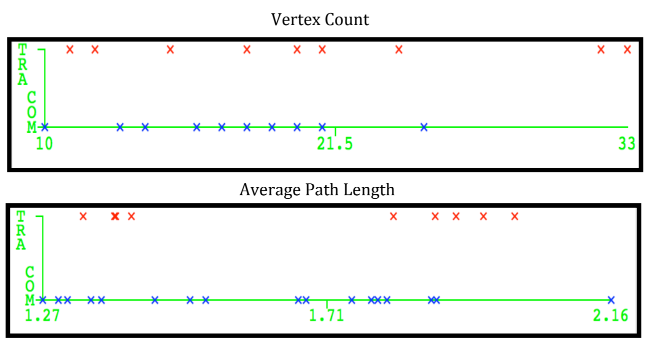

Figure 4.5: Above is a visualization of the vertex count distribution. Bottom is a visualization of the average path length distribution. Tragedies are in red and comedies are in blue in both visuals. PCA and Info. Gain chose average path length as the best feature. CFS chose vertex count.

[3] One-attribute classification experiments not described verified that average path length was indeed a better discriminator than vertex count.

4.2.2 Supervised Learning

Supervised learning was performed using the same techniques and methods described in 4.1.2 (same classifiers, same number of attributes, evaluation using ten-fold cross validation). The only difference was that histories were removed from the data set. Results were similar to the three-genre classification experiments with–again–the best method correctly classifying approximately 70% of the instances. However, the multilayer perceptron was not the best classifier in this experiment. It had potential to again be the best method, classifying 66% correctly using two attributes. But its accuracy fell by over 10% when the vertex count feature was dropped in the last trial. Conversely, *JRIP*, the rule learner, performed best by generating two rules: If *average path length* $\geq$ 1.817, then *class* = *TRAGEDY*. Else, *class* = *COMEDY*. JRIP'S jump to the forefront was unexpected going into the last trial since its accuracy was only 48% with two attributes and never reached 60% in the two other pervious trials. The 70% success rate is lower than the hypothesized accuracy for this two-class problem.

Comedy vs. Tragedy				
	Number of Attributes			
	7	3	2	1
J48 (Decision Tree)	62.96	51.85	48.15	55.56
JRIP (Rule Learner)	48.15	59.26	48.15	70.37
Clustering	37.04	33.33	40.74	33.33
Regression	48.15	55.56	55.56	59.26
Bayesian Network	62.96	62.96	62.96	62.96
Multilayer Perceptron	62.96	62.96	66.67	55.56

Figure 4.6: The accuracy rate of the six supervised learning methods with ten fold cross validation is presented above. The rule learner, *JRIP*, trained with one attribute, vertex count, performed best, correctly classifying 70.37% of the test instances.

4.3 Comedies vs Histories

The same experiment was repeated again, this time leaving out the ten tragedies. Since histories are similar to tragedies in plot, it was hypothesized that classification between these two classes would be similar to comedies vs tragedies.

4.3.1 Attribute Selection

With these two classes, there was no debate over which attribute would be the most powerful. All three chose vertex count to be the best attribute, supporting the initial hypothesis that histories can be characterized by their large cast of characters, a product of the plot usually revolving around the entirety of a royal court.

Method	Attribute Ranking
CFS Subset Eval	(1) Vertex Count, (2) Density, (3) Avg. Path Length
PCA	(1) Vertex Count, (2) Avg. Path Length, (3) Density
Info. Gain Ratio	(1) Vertex Count, (2) Density, (3) Avg. Path Length

Figure 4.7: Three attribute selection techniques performed on all seven attributes. Only the top three attributes are shown. *Vertex count* was selected as the most useful feature.

4.3.2 Supervised Learning

The six classification techniques were by far the most successful classifying histories vs. comedies. Every method achieved an accuracy rate greater than any other method in any other trial, the worst being classification via clustering's 81% with one attribute.

The *J48* decision tree correctly classified 96.30% of instances in every trial, making this technique the best in the history vs. comedy domain. The reason J48's success is uniform across the board is the learning algorithm's pruning feature. The same tree was used in all four trials. No matter how many attributes it was given, J48 pruned the tree down to only one level each time. The root node split on vertex count and classified networks with more than twenty-one nodes as histories and with less than or equal to twenty-one nodes as comedies, thus showing that vertex count alone is powerful enough to discriminate between these genres.

History vs. Comedy				
	Number of Attributes			
	7	3	2	1
J48 (Decision Tree)	96.3	96.3	96.3	96.3
JRIP (Rule Learner)	92.59	92.59	88.89	92.59
Clustering	74.07	74.07	74.07	81.48
Regression	92.59	92.59	92.59	92.59
Bayesian Network	92.59	92.59	92.59	92.59
Multilayer Perceptron	85.19	88.89	88.89	92.59

Figure 4.8: The accuracy rate of the six supervised learning methods with ten fold cross validation is presented above. The *decision tree* was best at classification with a success rate of 96.30% for every attribute subset.

4.4 Histories vs Tragedies

For the final time, the experiment was repeated again by removing comedies, leaving ten tragedies and ten histories. This classification task was hypothesized to be the hardest since histories and tragedies are the most similar in plot and content.

4.4.1 Attribute Selection

Attribute selection again chose vertex count to be the best attribute. Furthermore, CFS and Information Gain chose only vertex count. The other attributes provided zero discriminatory power according to these two methods. To confirm, PCA said the other attributes contribute to just 4% of the variance. The result aligns with the hypothesis that connectivity is similar in histories and tragedies since both genres feature malicious elements and factions in their plots.

Method	Attribute Ranking
CFS Subset Eval	(1) Vertex Count
PCA	(1) Vertex Count, (2) Avg. Path Length, (3) Density
Info. Gain Ratio	(1) Vertex Count

Figure 4.9: Three attribute selection techniques performed on all seven attributes. Only the top three attributes are shown. *Vertex count* was selected as the most useful feature.

4.4.2 Supervised Learning

Supervised learning provided surprising results. Histories and tragedies can be distinguished with a higher accuracy than comedies and tragedies, a result that conflicts with the hypothesis that comedies and tragedies are more dissimilar than histories and tragedies. Several learning methods achieved the highest level of accuracy, 80%.

Most notably, this experiment was the only one in which accuracy decreased with a decrease in attributes. J48, JRIP, and clustering all achieved their best performance with more than one attribute, showing that the 4% variance provided by the other attributes was helpful to some learning methods. While this result is surprising, it does demonstrate a flaw in PCA: variance does not always correlate with the feature that is most inherently useful to the classification task.

History vs. Tragedy				
	Number of Attributes			
	7	3	2	1
J48 (Decision Tree)	80	75	75	75
JRIP (Rule Learner)	70	80	80	75
Clustering	70	65	55	55
Regression	80	80	80	80
Bayesian Network	80	80	80	80
Multilayer Perceptron	70	70	75	80

Figure 4.10: The accuracy rate of the six supervised learning methods with ten fold cross validation is presented above. Every method except clustering achieved the highest accuracy level of 80%.

4.5 Errors, Improvements, and Future Work

A conspicuous flaw in the experiment is that only the *Hamlet* social network was verified for correctness. It was assumed that the automatic extraction method, since it was tested on one play, would work well enough on the thirty-six other plays. Yet it is unlikely that the extraction method produced a perfect network for every play, and therefore many of the structural features probably are not precisely correct. However, the error should be minimal since no instances seemed to have an unreasonable feature value and missing or invalid edges should make up only a small portion of a network's total edges, only skewing the structural features slightly.

Another (probably more detrimental) flaw is that there is significant sub-genre diversity within the three classes. For example, *The Taming of the Shrew* and *The Tempest* are both labeled as comedies; however, these two plays are very different in plot. The former strictly holds to the definition of a comedy from the modern period: a diverse cast squabbles for three acts and then ends up married at the end. The latter, on the other hand, is classified as a comedy because of its lack of atrocities and use of magical plot devices. Yet the play features little of the humor and farce that *Taming of the Shrew* does, and often modern critics call the play a 'tragicomedy' or a 'romantic drama.' Ambiguity also exists within the definitions of the tragedy and history genres: *Macbeth*, *King Lear*, and *Richard III* all feature rulers and fights for government power but only two out of the three are tragedies. Furthermore, each of these three plays is adapted from a historical account and describes excessive bloodshed, and thus it wouldn't be unreasonable for *Richard III* to be a tragedy or vice versa.

Besides checking each network for veracity, other improvements include testing a wider variety of supervised learning techniques and structural features. we tried to cover a wide range of features and techniques, but there is the possibility that better methods and network features still exist (such as the ones used in [GKS]). Classifier accuracy also can be increased through ensemble meta-algorithms (such as *bagging* or *AdaBoost*) or the combination of classifiers, as has been seen in numerous applications over the last two decades.

However, the lack of high accuracy (90%+) in every experiment except histories vs. comedies and the relative ineffectiveness of structural features besides *vertex count* at distinguishing genre may suggest that the supposition that genre is a function of connectedness and potential for information exchange throughout a social network is flawed or incomplete. One improvement could be incorporating time. The development of plot over time is an integral part of fiction, and perhaps genre could be better distinguished by tracking how the network evolves over time. Dynamic literary social networks have not been studied extensively [ACJR12].

4.6 Conclusions

Genre classification via the structural properties of a play's social network showed limited success. While histories and comedies can be distinguished with greater than 96% accuracy, the three-genre classification experiment, the most useful task, only achieved 70% accuracy. Furthermore, almost all of the error stems from misclassification of tragedies, as seen in the confusion matrix in Figure 4.11, meaning the social networks of tragedies showed almost no features that would discriminate them from the network's of comedies.

No supervised learning method was shown to be the best suited for this domain. Every method tried except for clustering achieved top accuracy in at least one experiment, with the decision tree, rule learner, and multilayer perceptron meeting the best performance in two experiments each. However, Bayesian Networks achieved the best general performance as calculated by accuracy rate averaged across all trials (74.77%), albeit regression and decision trees were above 73%. It is most likely that these averages speak to these method's ability to tolerate useless attributes more so than their capability in the domain.

```
=== Confusion Matrix ===

  a  b  c   <-- classified as
 10  0  0 |  a = HISTORY
  1 15  1 |  b = COMEDY
  3  7  0 |  c = TRAGEDY
```

Figure 4.11: A confusion matrix showing the classification results of the *J48* decision tree is above. As seen by examining the final row, tragedies are not distinguishable when the data set is comprised of all three genres.

As for attribute analysis, it was observed that a network's structural properties, such as density, diameter, average path length, max clique size, and main character PageRank, showed little discriminatory power between genres, accounting for only 4% of the data set variance (calculated by PCA) in the worst case. Vertex count, conversely, was shown to be capable enough to be used alone for supervised learning methods. The number of characters in a play was expected to be helpful since comedies usually feature smaller casts than other genres, but we did not predict that character count would be by far the most important feature. The most substantial result of this work is that plays with more than 21 characters are likely histories.

we believe the most significant impediment to the experiments described was the intra-genre diversity. Modern critics often argue over the correct classification of some of Shakespeare's plays, creating hybrid terms such as 'tragicomedy' and 'romantic drama.'

This variance within the classes is a fundamental flaw in the label definitions and probably caused the majority of the errors.

Given the learning methods' difficulties discriminating comedies from tragedies, most likely another method of analysis is needed to improve classification accuracy between these two genres. Preliminary results of experiments using sentiment analysis to classify comedies vs. tragedies shows more potential (see section 2.2), making genre classification via sentiment analysis a fertile field for future work.

Chapter 5

Sentiment Networks for Character-to-Character Analysis

Having shown the power of social network analysis and full-text SA, we now combine these methods. We believe a good model of literature lies in analyzing the flow of sentiment from character to character because fiction is fundamentally a description of character interactions. When SA is applied to a whole text, the interactions and character details are glossed over and missed. And when we use just the social network to analyze a work, emotion–that critical aspect of literature–is not captured. No previous work has combined these methods, giving edge weights to social networks via sentiment analysis.

5.1 Character-to-Character Sentiment Analysis

We mined for character-to-character sentiment by summing the valence values over each instance of continuous speech and then assumed that sentiment was directed towards the character that spoke immediately before the current speaker. Although this assumption doesn't always hold and is rather naive, we've found that the signal obtained from face-to-face dialogue drowns out, in most cases, the noise from when the assumptions do not hold. For example, Hamlet's sentiment rankings upon the conclusion of the play are shown in Figure 5.1. Not surprisingly, Claudius draws the most negative sentiment from Hamlet, receiving a score of -27. Gertrude receives the third most positive sentiment from Hamlet, which might be unexpected considering Hamlet thinks of his mother as an adulterer throughout the play.

Character	Hamlet's Sentiment Valence Sum
Guildenstern	31
Polonius	25
Gertrude	24
Horatio	12
Ghost	8
Marcellus	7
Osric	7
Bernardo	2
Laertes	-1
Ophelia	-5
Rosencrantz	-12
Claudius	-27

Figure 5.1: The characters in *Hamlet* are ranked by Hamlet's emotions towards them. Every time Hamlet speaks, his words' valence values are summed and assumed to capture his feelings towards the previous speaker. Claudius draws the most negative emotion from Hamlet and Guildenstern draws the most positive.

5.1.1 Peering into the Queen's Closet

To gain more insight into this mother-son (i.e. Gertrude-Hamlet) relationship, we examined how their feelings towards one another change over the course of the play. Figure 5.2 shows the results of dynamic character-to-character sentiment analysis on Gertrude and Hamlet. The running total of Hamlet's sentiment valence toward Gertrude is tracked by the black line, and Gertrude's feelings toward her son are tracked by the opposite boundary of the green-red area. The line graph shows a dramatic swing in sentiment around line 2,250, which corresponds to Act iii, Scene iv.

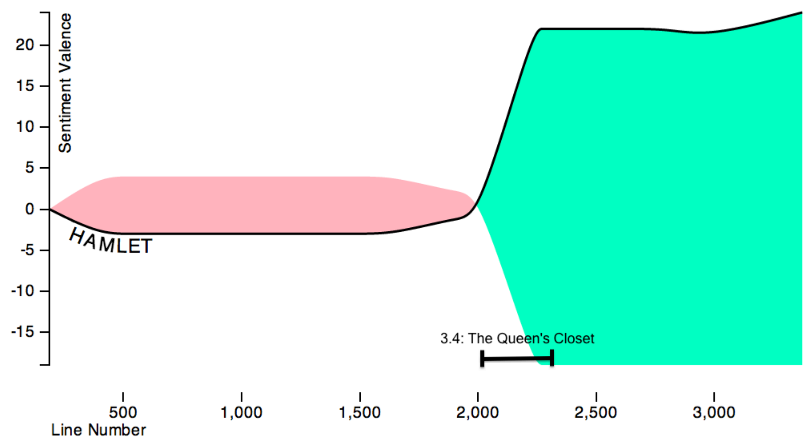

Figure 5.2: The above chart tracks how Gertrudes and Hamlets sentiment towards one another changes over the course of the play. Hamlets sentiment for Gertrude is denoted by the black line, and Gertrudes for Hamlet is marked by the opposite boundary of the red/green area. The drastic change in *ACT III SCENE IV: The Queen's Closet* is consistent with plot events since in this scene, Hamlet commits a murder before his mother and Gertrudes surprise at Hamlets accusations hints that she had no involvement in the plot to murder King Hamlet.

In this scene, entitled *The Queen's Closet*, Hamlet confronts his mother about her involvement in King Hamlet's death. Gertrude is shocked at the accusation, revealing she never suspected Hamlet's father was murdered. King Hamlet's ghost even points this out to his son: "But, look, amazement on thy mother sits" (3.4.109). Hamlet then comes to the realization that his mother had no involvement in the murder and probably married Claudius more so to preserve stability in the state. As a result, Hamlet's affection towards his mother grows, as exhibited in the sentiment jump from *-1* to *22*. But this scene has the opposite affect on Gertrude: she sees her son murder an innocent man (Polonius) and talk to an invisible presence (she cannot see King Hamlet's ghost). Gertrude is coming to the understanding that Hamlet is not just depressed but possibly mad and on a revenge mission. Because of Gertrude's realization, it is only natural that her sentiment undergoes a sharply negative change (*1* to *-19*).

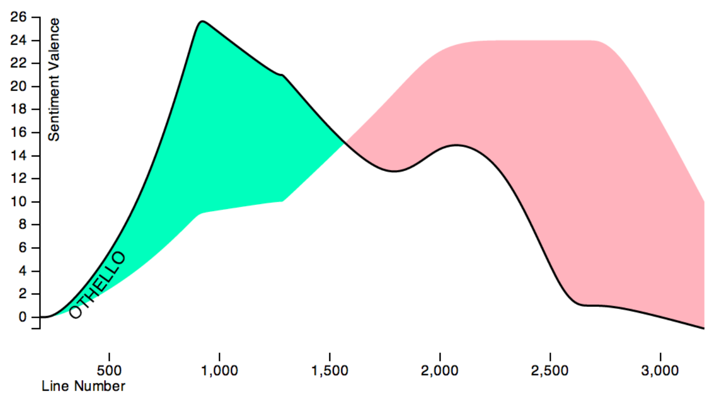

Figure 5.3: The above chart tracks how Othello's and Desdemona's sentiment towards one another changes over the course of the play. Othello's sentiment for Desdemona is denoted by the black line, and Desdemona's for Othello is marked by the opposite boundary of the red/green area. As expected, the line graph shows Othello has very strong positive emotions towards his new wife at the beginning of the play, but this positivity quickly degrades as Othello falls deeper and deeper into Iago's deceit. Also not surprisingly, Desdemona exhibits consistently positive feelings for her husband until the very end of the play when it becomes obvious he no longer trusts her.

Notice in this analysis of Gertrude's and Hamlet's relationship, sentiment analysis not only surfaced a possibly inconspicuous aspect of *Hamlet* (that Hamlet may care strongly for his mother) but also pinpoints a scene that drastically influences their relationship trajectories. While this case may be atypically revealing, it shows sentiment analysis' use to people reading and studying the play and possibly literary theorists looking for new insights.

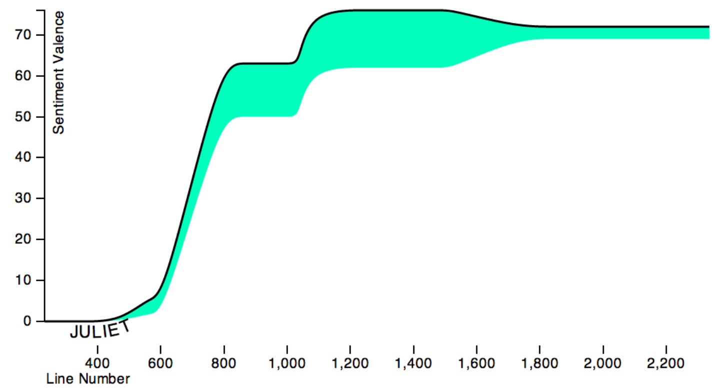

Figure 5.4: The above chart tracks how Juliet's and Romeo's sentiment towards one another changes over the course of the play. Juliet's sentiment for Romeo is denoted by the black line, and Romeo's for Juliet is marked by the opposite boundary of the green area. Aligning with our expectations, both characters exhibit strong positive sentiment towards the other throughout the play.

5.1.2 Analyzing Shakespeare's Most Famous Couples

After running this automatic analysis on all of Shakespeare's plays, not all the results examined were as enlightening as the Hamlet vs. Gertrude example. Instead, the majority supported our already held interpretations. We will now demonstrate what the technique revealed about three of Shakespeare's best known relationships. Figure 5.3 shows Othello vs. Desdemona sentiment dynamics. We clearly see Othello's love for his new bride climaxes in the first third of the play and then rapidly degrades due to Iago's deceit while Desdemona's feelings for Othello stay positive until the very end of the play when it is clear Othello's love for her has become poisoned. For an example of a contrasting relationship, Figure 5.4 shows Romeo vs. Juliet. As expected, the two exhibit rapidly increasing positive sentiment for each other that only tapers when the play takes a tragic course in the latter half. Lastly, Figure 5.5 shows Petruchio vs. Katharina (from *The Taming of the Shrew*). The phases of Petruchio's courtship can be seen: first he is neutral to her, then 'tames' her with a period of negative sentiment, and finally she embraces him, as shown by the increasingly positive sentiment exhibited in both directions.

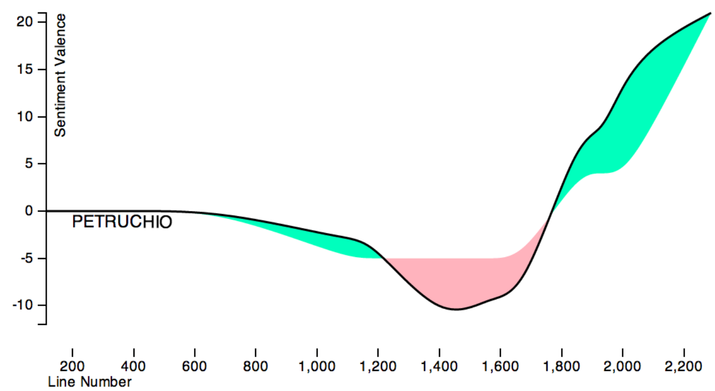

Figure 5.5: The above chart tracks how Petruchio's and Katharina's sentiment towards one another changes over the course of the play. Petruchio's sentiment for Katharina is denoted by the black line, and Katharina's for Petruchio is marked by the opposite boundary of the red/green area. The period from line 1200 to line 1700, during which Petruchio exhibits negative sentiment, marks where he is 'taming' the 'shrew.'

5.1.3 Is *King Lear* Shakespeare's darkest play?

According to Figure 2.1, *Titus Andronicus* is Shakespeare's most negative play. However, while that play may be extremely gruesome, there are still relationships that have relatively high valence scores. For example, Tamora has a valence score of 25 towards Saturninus. But when looking through the interpersonal relationships described in *King Lear*, the second lowest scoring tragedy in Figure 2.1, that play looks even bleaker. Only two relationships have a valence score over 11, Cordelia towards Lear (35) and Kent towards Lear (54). Lear, overall, is probably Shakespeare's darkest work since all characters have very few relationships above zero. Take the main character, King Lear, for example (Figure 5.6): he has four barely positive relationships and eight negative ones.

Character	Lear's Sentiment Valence Sum
Cornwall	9
France	5
Burgundy	4
Edgar	1
Edmund	-2
Cordelia	-3
Albany	-9
Regan	-12
Fool	-20
Goneril	-20
Gloucester	-38
Kent	-54

Figure 5.6: The characters in *King Lear* are ranked by Lear's emotions towards them. All of Lear's relationships are either negative or barely positive, which is common in this play.

Yet the exceptional positive relationships are worth noting. Cordelia shows unwavering positivity (+35) towards her father despite him banishing her and not returning the positivity (-3). Similarly, Kent feels extremely positive towards Lear (+54)–even though Lear banishes him as well–while, interestingly, Lear has the same valence magnitude towards him but with negative polarity (-54). Relationships unbalanced to this degree are not found in any other play, letting us propose that the steadfast loyalty Kent and Cordelia show towards Lear is the play's defining characteristic.

5.2 Structural Balance Theory

As shown in the previous section, the results of character-to-character analysis can be done visually, but we also wanted to explore the possibility that these networks could be explained by a general sociological model. We tested if Shakespeare's plays obey *Structural Balance Theory* (SBT), the idea that a friend of a friend is also your friend (i.e.

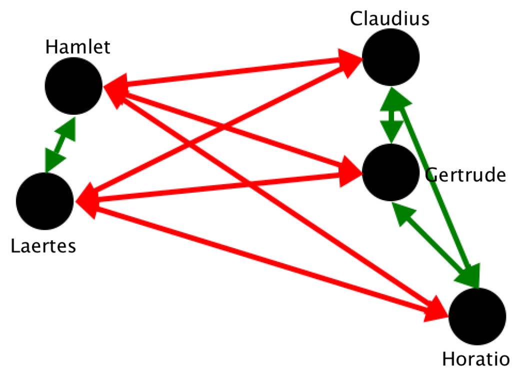

Figure 5.7: The above network shows Structural Balance Theory's prediction of how the characters in *Hamlet* will split allegiances when undergoing trauma. Green lines mark positive sentiment between characters, red lines mark negative sentiment. The network was obtained by repeatedly squaring the sentiment network's adjacency matrix until the polarity (sign) of edges stabilized.

transitivity of sentiment) [MKKS11]. This theory is captured by triangles in the graph. If two sides of a triangle have positive sentiment, the third side is likely to have positive sentiment or become positive over time, or vice versa for negative connections.

Marvel et al. showed that SBT can not only be modeled by a graph, but the evolution of relationships under stress can be simply modeled [MKKS11]. If X is an $n \times n$ adjacency matrix with nonzero entries that represent real-valued un/friendliness scores towards other nodes in the network, $\frac{dX}{dt} = X^2$. That is, repeatedly squaring the adjacency matrix simulates stress on the network, eventually causing the signs of X's entries to stabilize. Marvel et al. demonstrated that their method could accurately predict observed, real-world network fracturing. For instance, their method correctly predicted how countries sided during World War II.

We ran Marvel et al.'s procedure on the adjacency matrices of our Shakespearean sentiment networks to test if we could predict how a play's characters will split into factions just from looking at the state of the sentiment network after Act II. We saw varied results. Some plays converged to an arguably reasonable division. *Hamlet*'s results showed Gertrude, Claudius, and Horatio in one group, Hamlet and Laertes in an adversarial group (Figure 5.7), and Ophelia and Polonius as occupying middle ground, having positive and negative connections to both sides. But *Othello*'s matrix did not converge to a reasonable division. After every time step, Othello and Iago had a positive edge between them, which contradicts the enmity the reader knows Iago has for Othello. But since that hate is never reveled to Othello's face until the end of the play and Iago provides his commander with nothing but flatteries until that point, the sentiment analysis does not and cannot detect the deceit. This problem shows a clear inadequacy in our shallow SA methods. Overall, we did not find the factions calculated to be compelling enough to support the idea that Shakespeare's plots can be accurately modeled mathematically by SBT.

5.3 Sentiment Network Web Visualization

To facilitate exploration of the results of SA and social network extraction on all of Shakespeare's works, a web visualization was created with the D3 JavaScript library. It can be viewed at

$$http://www.lehigh.edu/\sim etn212/ShakespeareExplorer.html$$

To initiate use, a play is selected. Clicking on any of the thirty-six plays reveals that play's social network. Mousing-over any node in the network fades all nodes that don't share an edge with the selected node and highlights all non-faded nodes in either red, if the selected node (character) has negative sentiment towards that node, or green if positive sentiment. Clicking on the selected node shows the edge weights on the connections to the unfaded nodes, producing a ranking like the one seen for Hamlet in Figure 5.1. Finally, selecting a name from the ordered list produces a difference line graph (see Figure 5.2 for an example) showing the dynamics of the character pair's relationship.

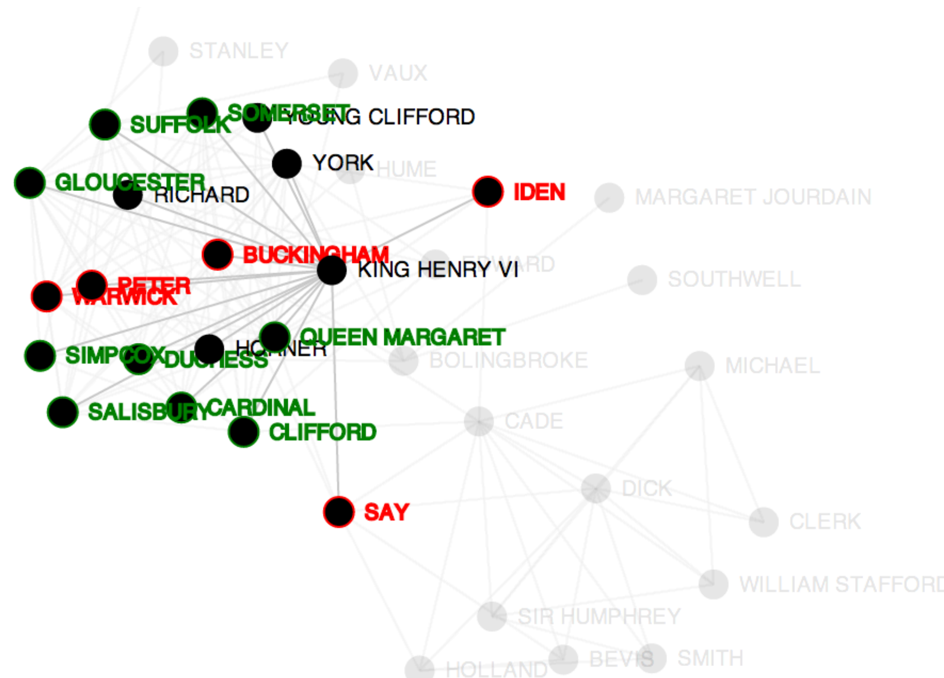

Figure 5.8: The state of the JavaScript visualization when *Henry VI Part 2*'s namesake character is selected.

Chapter 6

Future Work and Conclusions

6.1 Inadequacies of Shallow Sentiment Analysis

Given the problems with sentiment analysis seen first hand in our SBT experiments, we looked for some way to identify subtle rhetorical elements such as irony and deceit. We wanted some shallow method that could identify that Iago is lying and does not have the positive sentiment evident from superficial analysis.

Psychologist James W. Pennebaker has had some success in this area. By studying the rhetorical tendencies of individuals in various mental states, he pinpointed several shallow linguistic features or patterns that can reveal deeper emotions. For example, Pennebaker compared debates in which students supported an argument that they agreed with and one that they did not agree with, essentially lying in the latter case. The truthful debates showed an increased number of self references, adjectives, and non-negative words when compared to the untruthful debates [NPBR03]. He reasoned that liars tend to distance themselves from their falsehoods (lack of self reference) and provide less details since lying is cognitively more challenging than telling the truth.

Although Shakespeare's characters and their emotions are artificial, artistic constructions and thus not likely to reveal psychological 'tells' the way Pennebaker's specimens did, we tested Shakespeare's plays for Pennebaker's markers of deceit, hoping the analysis would show Shakespeare's villains as more untruthful than the heroes. We analyzed Shakespeare's three most famous tragedies: *Othello*, *Hamlet*, and *Macbeth*. For each character who spoke more than 900 words, we averaged the percentage of their words that were self-references (I, me, myself, etc.), that had a non-negative valence value, and that were adjectives or adverbs (to capture descriptiveness). The resulting percentages were then normalized for each play, forcing the 'most truthful' character's score to 100 and the 'most deceptive' character's to 0. The results are shown in Figure 6.1.

Pennebaker's features produced an ordering that is consistent with readings of these tragedies. Iago, Emilia, Claudius, and Lady Macbeth all were determined to be the least truthful in their respective plays. Emilia may seem to be an outlier here, but actually she tells a significant lie about the origin of the handkerchief, the most influential piece of 'evidence' in convincing Othello of his wife's infidelity. While these rankings alone are probably not enough to discriminate heroes from villains–as Horatio, who is certainly not a villain, is near the bottom of *Hamlet*'s rankings–there does seem to be some worth in the analysis. An alternate explanation for why the Pennebaker analysis seems to work is that the villains, at least in the case of *Hamlet* and *Macbeth*, are heads of state and thus would be inclined to use *we* (in reference to the state) in their speeches.

Character	Truthfulness Score
OTHELLO	
1. Desdemona	100.00
2. Cassio	62.50
3. Othello	37.50
4. Emilia	0.00
5. Iago	0.00
HAMLET	
1. Polonius	100.00
2. Ophelia	81.25
3. Gertrude	68.75
4. Hamlet	62.50
5. Laertes	56.25
6. Horatio	37.50
7. Claudius	0.00
MACBETH	
1. Macduff	100.00
2. Malcolm	73.68
3. Macbeth	15.79
4. Lady Macbeth	0.00

Figure 6.1: Above is the chart showing the truthfulness rankings (as determined by Pennebaker's features) for Shakespeare's three most famous tragedies. The rankings were determined by averaging the percentage of self-references, non-negative words, and descriptors a character uses: the higher the average, the more truthful a character is. The results have been normalized, forcing the most truthful character's score to 100 and the least truthful's score to 0. The character rank-order aligns with popular interpretations of the three plays.

6.2 Extending to Unstructured Texts

While we performed experiments on just Shakespeares plays, note that the described character-to-character SA technique should be extensible to any work of fiction written since the Elizabethan Period. The sentiment lexicon we used, AFINN, is designed for modern language; thus, it should only provide better analysis on works written after Shakespeare's. Furthermore, character-to-character analysis should be able to be applied to novels (and other unstructured fiction) if Elson et al.'s speaker attribution technique is first run on the work [EDM10].

Not only can these techniques be extended to novels but also be made more precise. For instance, the assumption that the current speaker's sentiment is directed toward the previous speaker is rather naive. A speech could be analyzed for context clues that signal that the character speaking is not talking about someone present but about someone out of the scene. The sentiment could then be redirected to the not-present character.

6.3 Conclusion

As demonstrated, basic, un-customized sentiment analysis can be used with some success to recognize and analyze human relationships described within a document and output an interpretation that matches human expectations. Word valence is clearly linked to document genre. Yet more importantly, sentiment can be tracked over the course of a work's plot to pinpoint influential moments (as is the case with Hamlet's Act 3, Scene 4) as well as look for possibly hidden insights. For instance, Hamlet's devotion to his mother is a facet of *Hamlet* we have never noticed. Despite some success, sentiment analysis showed a clear flaw. Deceit, an essential and pivotal element of many literary works, is hardly detectable through sentiment lexicons. Psychological models have made the most progress identifying suppressed emotions in words, but we doubt that these models are a general solution.

To judge our work's steps towards automatic, deep literary analysis or towards a use in automated, machine reading, more work must be done. Firstly, a shallow method for detecting subtle rhetorical features (irony, humor, deceit, etc.) must be better studied before standard sentiment analysis becomes robust and fully reliable. Moreover, more experiments on more types of literature are needed. Our next step will be to run character-to-character sentiment analysis on novels by using a speech attribution method similar to Elson et al.'s [EM10].

Bibliography

[ACJR12] A. Agarwal, A. Corvalan, J. Jensen, and O. Rambow. Social network analysis of alice in wonderland. *NAACL-HLT 2012*, page 88, 2012.

[EDM10] D.K. Elson, N. Dames, and K.R. McKeown. Extracting social networks from literary fiction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 138–147. Association for Computational Linguistics, 2010.

[EM10] D.K. Elson and K.R. McKeown. Automatic attribution of quoted speech in literary narrative. In *Proceedings of AAAI*, 2010.

[GKS] Sebastian Gil, Laney Kuenzel, and Caroline Suen. Extraction and analysis of character interaction networks from plays and movies.

[JA06] Mohsen Jamali and Hassan Abolhassani. Different aspects of social network analysis. In *Web Intelligence, 2006. WI 2006. IEEE/WIC/ACM International Conference on*, pages 66–72. IEEE, 2006.

[KH04] Soo-Min Kim and Eduard Hovy. Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[KNS$^+$08] J. Krauss, S. Nann, D. Simon, K. Fischbach, and PA Gloor. Predicting movie success and academy awards through sentiment and social network analysis. In *ECIS European Conference on Information Systems*, 2008.

[MKKS11] S.A. Marvel, J. Kleinberg, R.D. Kleinberg, and S.H. Strogatz. Continuous-time model of structural balance. *Proceedings of the National Academy of Sciences*, 108(5):1771–1776, 2011.

[Moh11] S. Mohammad. From once upon a time to happily ever after: Tracking emotions in novels and fairy tales. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 105–114. Association for Computational Linguistics, 2011.

[Mor] Franco Moretti. Network theory, plot analysis.

[MPNdL10] Rafael Maranzato, Adriano Pereira, Marden Neubert, and Alair Pereira do Lago. Fraud detection in reputation systems in e-markets using logistic regression and stepwise optimization. *SIGAPP Appl. Comput. Rev.*, 11(1):14–26, June 2010.

[Mut04] P. Mutton. Inferring and visualizing social networks on internet relay chat. In *Information Visualisation, 2004. IV 2004. Proceedings. Eighth International Conference on*, pages 35–43. IEEE, 2004.

[Nie11] F. Å. Nielsen. Afinn, March 2011.

[NPBR03] M.L. Newman, J.W. Pennebaker, D.S. Berry, and J.M. Richards. Lying words: Predicting deception from linguistic styles. *Personality and Social Psychology Bulletin*, 29(5):665–675, 2003.

[PBMW99] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: bringing order to the web. 1999.

[PGMJ11] Barbara Poblete, Ruth Garcia, Marcelo Mendoza, and Alejandro Jaimes. Do all birds tweet the same?: characterizing twitter around the world. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 1025–1030, New York, NY, USA, 2011. ACM.

[PL08] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135, 2008.

[PLV02] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

[PnH10] Anselmo Peñas and Eduard Hovy. Filling knowledge gaps in text for machine reading. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 979–987, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[Rom94] T. Rommel. so soft, so sweet, so delicately clear.a computer assisted analysis of accumulated words and phrases in lord byron's epic poem don juan. *Literary and linguistic computing*, 9(1):7–12, 1994.

[Smi03] A.E. Smith. Automatic extraction of semantic networks from text using leximancer. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Demonstrations-Volume 4*, pages 23–24. Association for Computational Linguistics, 2003.

[SV04] C. Strapparava and A. Valitutti. Wordnet-affect: an affective extension of wordnet. In *Proceedings of LREC*, volume 4, pages 1083–1086, 2004.

[TM00] K. Toutanova and C.D. Manning. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13*, pages 63–70. Association for Computational Linguistics, 2000.

[TM08] Ivan Titov and Ryan McDonald. Modeling online reviews with multi-grain topic models. In *Proceedings of the 17th international conference on World*

Wide Web, WWW '08, pages 111–120, New York, NY, USA, 2008. ACM.

[TNKS09] Tun Thura Thet, Jin-Cheon Na, Christopher S.G. Khoo, and Subbaraj Shakthikumar. Sentiment analysis of movie reviews on discussion boards using a linguistic approach. In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, TSA '09, pages 81–84, New York, NY, USA, 2009. ACM.

[Wil64] G. Williams. The comedy of errors rescued from tragedy. *Review of English Literature*, 5:63–71, 1964.

[ZZ07] Y. Zhao and J. Zobel. Searching with style: Authorship attribution in classic literature. In *Proceedings of the thirtieth Australasian conference on Computer science-Volume 62*, pages 59–68. Australian Computer Society, Inc., 2007.

Vita

Eric Thomas Nalisnick was born on February 6, 1990 in Spangler, Pennsylvania to Thomas and Regina Nalisnick. He was raised in Ebensburg, Pennsylvania and graduated Central Cambria High School in May 2008. In September 2008, he enrolled at Lehigh University to study English and Computer Science. Graduating *summa cum laude*, he was awarded a Presidential Scholarship which allowed him to continue his studies at Lehigh in pursuance of a Master's degree in Computer Science. He expects to graduate in May 2013 and pursue a PhD in Computer Science at the University of California, Irvine.