2014

University of North Carolina Wilmington

Master of Science in

Computer Science and Information Systems

Proceedings

https://csbapp.uncw.edu/mscsis

LITERARY ANALYSIS TOOL:
TEXT ANALYTICS IN A CREATIVE WRITING CONTEXT

Austin Grimsman

A Capstone Project Submitted to the
University of North Carolina Wilmington in Partial Fulfillment
of the Requirements for the Degree of
Master of Science

Department of Computer Science
Department of Information Systems and Operations Management

University of North Carolina Wilmington

2014

Approved by

Advisory Committee

Dr. Ron Vetter	Dr. Curry Guinn

Dr. Douglas Kline, Chair

Abstract

In the creative writing process, one of the most serious bottlenecks is obtaining reliable and consistent review. Gathering useful human-generated feedback is time and effort intensive. An automated method could greatly streamline the writing process, even if the feedback generated is not as robust as that created by human understanding, which is unfeasible to replicate. Limited but instantaneous feedback could accelerate the process of review and revision for a writer. In this paper, a user-friendly web app called the Literary Analysis Tool is described, designed to assess the changing emotional content of narrative prose, along with other properties of the text. This tool was demoed for a small audience of writers, and their feedback suggests the potential for forms of automated writing assessment oriented to assist a creative writer in assessing the content of their story in objective terms.

1. Introduction

The field of Creative Writing is diverse and difficult to quantify. With low barriers to entry and imprecise standards of quality, analyses and assessments of written prose tend to be highly subjective in nature. While writing can benefit from research, planning, and a careful analysis of prior work, ultimately, the only tools a writer needs are an implement to write with and, in the words of Stephen King, "a door which you are willing to shut." At present, there is no array of tools necessary in producing a quality first draft; only a skill-set to develop.

The second draft, however, is, in some cases literally, another story. Writing requires a closed door, but rewriting mandates an open one. A finished product can be created in a vacuum, but as a rule, all writers, no matter how experienced, benefit from reviewers offering fresh perspective. Readers are necessary for the creation of quality prose, for reasons as complex as writing itself.

A writer's perception of their work is colored by everything that went into creating it. They don't need to have their story told to them. They don't require a primer on the world in which the story takes place. They already know the characters better than anyone in their potential audience, without reading a single word. Everything they proofread is contextualized within a framework of the story they meant to write, and they use this information to fill in every gap and hide every flaw.

Simply, test readers are the only real way for a writer to know what they have written, rather than what they thought they wrote.

Furthermore, a fresh perspective can provide a writer with inspiration, either for revision or subsequent works. Review is not limited to analyzing a paper and identifying flaws, but

synthesizing the parts into a cohesive whole in their minds, drawing connections, and discovering aspects that catch their interest, intended or not. If readers consistently recognize positive traits in a narrative work, which a writer either did not intend, or did not consciously realize they intended, these traits are not invalidated by this lack of intentionality. The text which exists, exists independently of the text which was intended.

A clever writer should capitalize on such accidental (or unconscious) successes whenever possible. Similarly, if elements are discovered which are undesirable, it is important that they be recognized as quickly and as consistently as possible, so they can be removed or altered. This can't happen if reviews are careless, few in number, or inconsistent. Only when presented with a clear picture can a writer make an informed decision.

A writer may consciously choose to ignore a review, or even a consensus of reviews. This is often the correct choice, since popular writing does not necessarily correlate with good writing. However, if a writer does choose to persist with something many readers find undesirable, this should be the result of careful deliberation, not accident.

The services of a skilled reader are not easy to come by. Producing a meaningful review requires effort, patience, enthusiasm, and experience. Counter intuitively, skill as a writer does not necessarily translate into skill as a reader. In fact, in academia, literary criticism is a field of study largely separate from creative writing. In theory, a writer may not desire so thorough an analysis, unless their target audience includes academic fields. However, inexperienced readers tend to provide feedback limited to their gut impressions, often without much explanation. This also varies greatly from individual to individual.

This variability is itself another complication in the effort to produce actionable feedback on a piece of narrative prose. Literary quality remains nearly impossible to define. A reader's reaction hinges greatly on personal preferences. Human readers have favorite genres, favored tropes, character traits, writing styles, story structures, atmospheres, and a variety of traits so as to make any response as much an analysis of their personality as the story. These do not remain consistent even for a single individual. Tastes change, interest becomes saturated, and moods change from hour to hour. Consistency in review work is at best an elusive property.

1.1 Solution Definition

With modern text analysis techniques, it should be possible to provide a degree of automation in this process. This would, by necessity, be to a limited degree. It is not possible at this time to fully replicate the services of a human reader, because this requires parsing and understanding the meaning of a work as a whole. This is not to say that no useful automation can be implemented.

> "No computer in existence is capable of generating a critical paper on gender dynamics in King Lear, or the figure of the flaneur in the nineteenth-century French novel, or even the assonantal structures of Old German poetry. We would be within our rights to consider such feats among the hardest of the AI-hard problems now extant. To think of this as what literary critics do, however, is to state in broadly elliptical terms what is in fact only the end product of a process that shares some enormous similarities with the methods of the ludic tradition." [9]

Given the limitations imposed by the need for a human reader, even partial automation would be of benefit to a writer. In contrast to the lengthy process required to bring a work to a state of relative completion, find a suitable reader, persuade him or her to analyze the text, and produce a response, software could provide feedback in moments. In addition, this feedback, however limited, would be impartial and repeatable.

To maximize ease of use, this could best be implemented in the form of a web app. This would provide easy and installation-free access to hypothetical feedback generation utilities to any user regardless of platform.

For the purposes of this paper, the utility in question will be called the Literary Analysis Tool, implemented in JavaScript.

2. Review

2.1 Creative Writing

2.1.1 Overview

Evidence exists that the human thought process is strongly influenced by language learned early in an individual's development. Preverbal children develop concepts and spatial reasoning, but language determines what categories those concepts fit into, which influences what sorts of distinctions individuals become more or less sensitive to later in life. Concepts which are easily differentiated by language remain more distinct in an adult's thought process, and vice versa.

"Whereas adults maintain sensitivity to acoustic and mechanical distinctions that are captured by their native language, they decline in sensitivity to acoustic and mechanical distinctions that are not captured by that language."[11]

The invention of writing cannot be accurately dated. It most likely arose from techniques for recording agricultural harvests. Written storytelling arose from oral storytelling, and has similarly nebulous origins.

The importance of writing, recordkeeping, and language on the development of human civilization is far too broad and extensive to be detailed to any meaningful degree in this paper. By the mid 1450s, movable-type printing presses allowed for the creation of books in large numbers, allowing for the spread of literacy in the European populace. While no one factor can be said to have brought about the Renaissance, it would have probably been impossible without this invention.

The modern novel was developed through the 16th and 17th centuries from earlier novellas and romances. A novel is typically defined as a complex fictional narrative of 40,000 words or more, though there is no universally accepted standard.

The field of literary writing is typified by its lack of universally accepted standards. There is little practical description of what makes a novel, or other piece of literary fiction, beyond the most fundamental basics of words and sentence structure. Even this is occasionally disregarded. The study of it, however, falls into two fields of thought: Creative Writing, the training or guided development of new writers, and Literary Criticism, the analysis and interpretation of literary works. These fields display a surprising lack of overlap, with prospective creative writers taught little of the theoretical underpinnings of the techniques of literary criticism to be used in assessing their work.

Writing is judged in two ways: First, by the quality of its prose, the ability of a writer to create images in a reader's mind, to convey mood, to turn a phrase, and to communicate with tight, precise language. Second, by the quality of the story told, its adherence to structure, rising and falling action, character arcs, and continuity.

There currently exists no scientific means of quantitatively assessing a work of narrative prose. There are no objective metrics which could be taken and implemented, or techniques that can be plausibly automated. Those techniques which do exist to judge a work require much the same thought process that was required to write it in the first place, analyzing concepts such as characterization, character growth, and the development and conclusion of plotlines.

Assessing properties such as mood, the emotional attitude displayed by a work, and imagery, the ability of a work to paint an image in all five senses in a reader's mind, may prove

less arcane, as diction plays a vital role. Stories with a dark mood will contain words associated with dark moods, while stories with romantic overtones will adopt similarly appropriate diction.

While understanding of a piece is necessary to interpret the metaphors and similes typically used to evoke imagery, these tools tend to adopt specific forms of diction. Word choice plays a vital role.

2.1.2 The Writing Process in Overview

The process of creative writing can be roughly described in five phases: Prewriting, Drafting, Revising, Editing, and Finalizing.

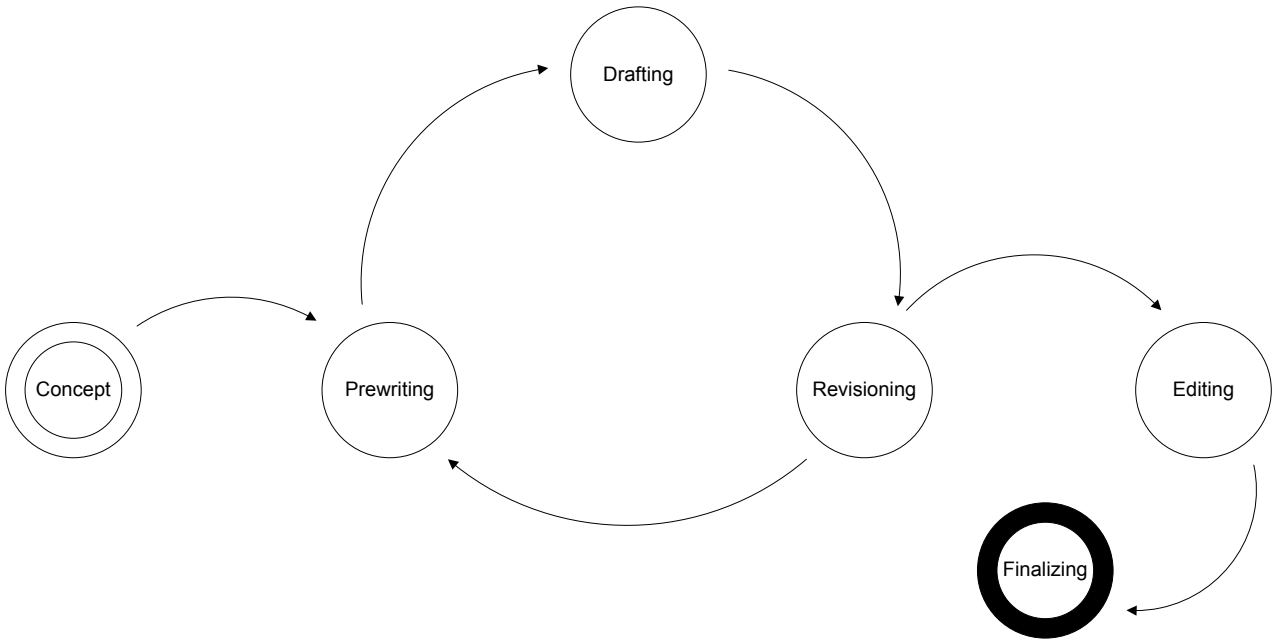
Prewriting is the planning stage in creating a work. This could be anything from a vague idea and a character name, to a comprehensive framework of notes and charts detailing the story's structure, its subplots, its characters and their arcs, to then be fleshed out and given life with prose.

Drafting is the initial act of actually writing a story. This is where the meat of the work is first created. Here is the most commonly presented act of writing, with an author typing out sentence after sentence to produce readable narrative prose.

After a draft is made, it can be revised. The manuscript is read and re-read, with alterations ranging from suboptimal wording to entire passages being written, deleted, or rearranged.

Editing is the act of touching up a work to make it more readable and presentable. It consists of alterations made at a finer grain. Grammar, punctuation, and spelling are made correct.

Finalization is the cessation of these activities. The author and/or editor work to decide what parts are good enough, and eventually, that the work has reached a stopping point. Honing a piece of writing to perfection is a process with no end, so the key, at this stage, is the skill, experience, and mental discipline to know when is enough.



The archetypal writing cycle

Much like software engineering, there are many different ways to move through these phases. Generally, the method is cyclical, with iterations of proofreading, revisions, alterations to the plan, and (ideally) forward progress through the trajectory of the plot, tying a predetermined beginning to a predetermined end, or allowing the story to branch from plot point to causal plot point to reach some satisfactory conclusion.

Even this loose structure is only a guideline. Prewriting, drafting, and revision can often blur together. Entire projects can be abandoned and used as material for later work. Free-writing

can be created in one pass, and a framework of notes only later built around it. It's a rare author with the discipline to write a draft without any changes to the original plan.

However, a lack of structure in the writing process can lead to paralysis, wasted time, or a lack of focus in the finished product. Moving too freely between phases is counterproductive. Just as it takes discipline to know when a work has reached a stopping point, a writer must also know when to, at least roughly, halt one phase and move on to the next. If an author revisits decisions made in the prewriting phase too freely, the story may lack the structure to maintain coherence. If revising tends to mean adding fresh text at every opportunity, it will quickly become long and rambling. If the author keeps too much of an eye towards revising, it becomes difficult to efficiently and fluidly draft a work at all.

On a more abstract note, a writer must maintain a balance of confidence. The most common pitfalls for prospective writers are to either possess too little faith in their own abilities and the quality of their ideas to actually produce a finished product, or to overestimate themselves to the point of being completely immovable by criticism, preventing themselves from correcting mistakes or improving their skill in the long term.

2.1.3 Literary Criticism

This encompasses the categorization, interpretation, evaluation, and general study of literature. It is considered a separate field from creative writing, and more closely linked to philosophy. While creative writing focuses on the act of writing and the training of prospective writers, literary criticism seeks to unlock the meanings behind a piece of existing writing, studying aspects of it that the writer may not have intended, or even necessarily been aware of.

This might take place in the form of broad, sweeping interpretations of themes and meanings, or in the form of close reading. A close reading is a highly detailed analysis of a brief passage. This seeks to exhaustively analyze the underlying meanings behind said passage, and how it relates to its context in the work, and beyond. A close reading might put more time and thought into a few paragraphs, or even a few sentences, than the author put into writing them, and might even succeed in unlocking genuine underlying meaning not obvious to even an alert reader.

Literary theory is a philosophical tool linked with literary criticism. It defines a particular lens, or lenses, through which a particular work, or larger body of works, may be interpreted. These are often mutually exclusive. For instance, psychoanalytic literary theory uses a work as a means to analyze the author's personality and thought processes, while New Criticism attempts to completely separate a work from its author, analyzing only the text as presented while deliberately ignoring information not contained within it, such as the background of its creation, or author intent.

In this sense, New Criticism may be considered the subfield of literary criticism closest to the focus of this project. However, this does not pose nearly a sufficient amount of similarity to prove useful for the fields of Text Analysis, Natural Language Processing, or Sentiment Analysis. Analyzing a work independent of the broader context of an author's career represents the default functionality of existing automated processes not specifically designed to do otherwise.

2.2 Text Analytics

2.2.1 Overview

The term Text Analytics refers to an array of techniques intended to automatically distill, organize, and store the content of a body of text. It is a broad field taking input from any studies which may prove useful, from computational linguistics, to statistical analysis, to machine learning. Also known as Text Mining, this is considered a subset of Data Mining, and inherits many techniques from that overarching field.

The process of Text Mining consists of the retrieval of unstructured text, extraction of information, the assembly of semantic metadata resulting in structured content, and through this, knowledge discovery. In this way, implicit knowledge may be converted to explicit knowledge. [3]

There is little that can universally be said about the specific techniques composing Text Analytics. The field includes, essentially, any method that can be used to organize text into table-based results, no matter how ad-hoc, according to any criteria or organizational methodology no matter how arbitrary. The uses of Text Analytics are essentially without limit, and as such there are uncountable goals and purposes which can be achieved in just as many ways.

For the purposes of this document, the terms Text Analytics, Text Mining, and Text Analysis are to be considered interchangeable. No widely agreed differentiation between these terms appears to exist.

At present, it is impossible for computers to be taught to truly parse and understand the meaning behind text, as a human would. However, there are a plethora of statistical techniques to

examine its content. Word and phrase usage may be counted, aggregated and categorized according to predetermined rules and definitions.

Text Analytics can be seen as a search engine in reverse. Instead of entering search terms and sorting through a body of text to find relevant passages, the entire corpus is analyzed to find and present the most prevalent or important terms. Searching is a "top down" means of organizing data, taking requirements and producing best-fits. Text analytics, on the other hand, is "bottom up," taking the data itself as the prime mover of the process, producing outputs without any predetermined 'leads' for the program to follow.

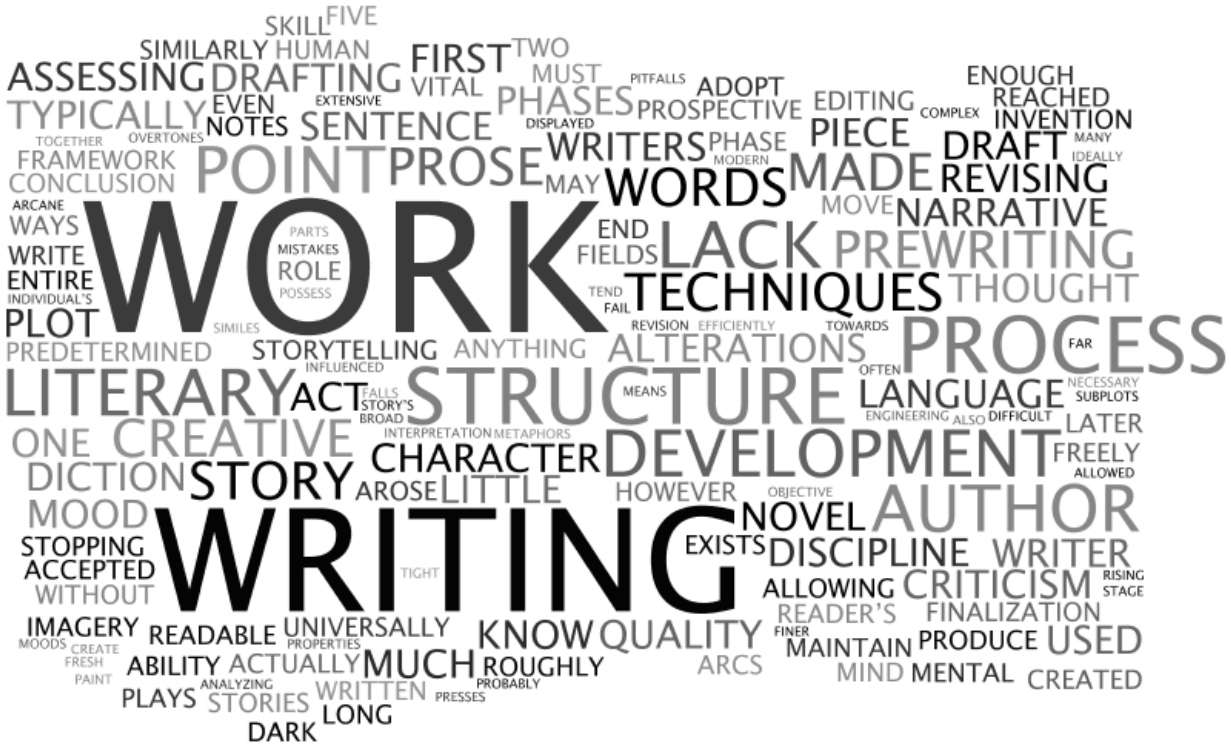
Text Analytics are most commonly used to analyze a large corpus, containing many documents, for the purpose of retrieving information useful to a business. The widespread growth of personal computing and social networking, as well as increased rates of publication, has resulted in reams of new text being generated daily, orders of magnitude beyond what any individual or realistically sized organization can read or analyze in any meaningful way.

Nevertheless, businesses must base their decisions on quantifiable patterns in their customer base and the business environment at large. Governments must adapt to ever larger amounts of data, most of it meaningless, within which could exist information vital to national security. Only automated text analysis can provide the sheer throughput of analysis necessary to even point human analysts in the right direction.

An additional advantage of Text Analytics over human readers is impartiality. In many cases, humans produce a sufficient amount of bias as to greatly reduce the obvious advantages of an intelligent reader. One reader comparing several passages and generating a report can provide

useful data, but standardization across many reviewers/analysts can prove difficult to impossible. A given algorithm will generate the same result from the same data each time.

The subset of Text Analytics most relevant to this project is Sentiment Analysis. This covers a range of techniques and tactics for automatically discerning human attitudes from a given body of text. Also of note is Opinion Mining, which automatically determines views are being expressed, typically positive or negative, about a predetermined subject. [6, p.7]



2.2.2 Implementation

The archetypal implementation of Text Analysis is the word count. This is a relatively simple task, where a document is parsed into its component words, and statistical data is

gathered regarding which words are used most frequently. The more commonly a word is used, the more prominent and generally more important it is, allowing the subjects of documents to be quickly determined with reasonable accuracy.

Unfortunately, this process can quickly be confounded by the use of synonyms and abbreviations, pushing key terms below detection thresholds, or homonyms artificially inflating the importance of other terms. At the same time, glue words must be filtered out, or risk "the" being the most repeated and thus most important word in any document.

Pronouns should be assigned meaning, as best as can be managed. The attribution of non-proper nouns is itself a large and confounding problem. The more accurately non-proper nouns can be linked with their meanings, the better Text Analytical techniques will function. However, in written English, this can be difficult or impossible to determine. Many sentences require context to determine the meaning of, and many more have multiple interpretations which not even humans can determine the meaning of with absolute certainty. As such, pronouns are nearly impossible to accurately attribute for text analysis purposes.

From word counts, a wide array of techniques present themselves, from complex and painstakingly coded rule-based techniques, to learning algorithms, such as Bayesian filters, which learn commonly reoccurring words and phrases and gauge the probability of a certain item fitting certain criteria, accepting/blocking those which reach a preset threshold.

Even assuming an effective and competent implementation, word count-based methods of sentiment analysis are fraught with pitfalls. Since these function without regard for the grammatical syntax of human language, the meaning of many opinions cannot be determined solely from emotionality of word choice. Modifiers frequently reassign meanings mid-sentence.

Positive words can be used to describe what a subject lacks, or vice versa. More creative writers can express strong sentiments with a balanced word choice, or even using words not assigned any negative or positive emotionality whatsoever. [6]

2.2.3 Computational Linguistics/Natural Language Processing

Related to Text Analytics is the field of Computational Linguistics. Like Text Analytics, this is an interdisciplinary field covering a wide array of techniques related to the automated processing of language.

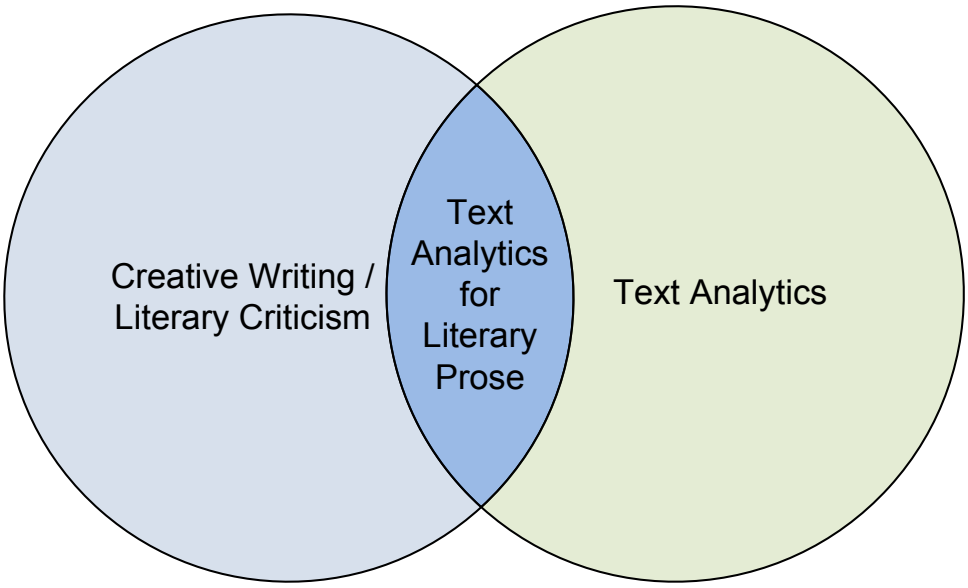
Computational Linguistics differs, however, by focusing on sentence structure, and the interrelationship those sentences have with one another. The goal here is not merely to draw conclusions about the text based on a preponderance of certain words or the length of sentences, but to interpret the meaning behind the text. Natural Language Processing is considered either an alternative word for Computational Linguistics, or a subset thereof. This differs further from Text Analysis by considering the spoken word, including turning audio into text and text into audio.

Computational Linguistics reaches back to the dawn of the modern computing age. Much like artificial intelligence, it was initially assumed in the mid 1950s that automated processing of language would be achieved with a minimum of fuss, leading to machine translation, and so on. This goal has proven elusive.

The internal divisions of this field are, at present, not sharply defined. Likewise, the dividing line between Natural Language Processing and the remainder of the field of Computational Linguistics is not easily established, nor is the point where Text Analytics ends

and other forms of automated reading begin, including Computational Linguistics and other forms of Data Mining, are not rigorously defined. While Computational Linguistics can be more easily categorized into standardized theories, Text Analysis tends to be more fluid, with techniques being chosen or discarded depending on the corpus being analyzed, and the nature and desired formatting of information.

2.3 Existing Intersections of Text Analytics and Creative Writing



2.3.1 Overview

Meaningful bridging of the gap between traditional Text Analytics and traditional Literary analytical techniques must begin extend from the former field. No extant methods native to the fields of literary criticism or creative writing can be meaningfully implemented as a computer system. All relevant techniques are reliant on a human to interpret comparably rough

guidelines, and draw on their personal experience as a writer or a reader, with which they can draw an informed opinion on the work in question.

Automated analysis of writing exists, but is exclusively or almost exclusively in the form of simple scoring functions for reading level, with which a work may be categorized by difficulty.

One popular related form of research is Topic Detection. While different from theme or mood detection, topic detection can be considered a parallel form of research, automatically analyzing traits present in a given sample of text. Significant work has been done on Topic Detection. [4]

A closer fit is Sentiment Analysis, also called Sentiment Classification, a subtopic of Text Analysis research. Sentiment Analysis seeks to automatically detect and categorize opinions, and whether the writer's reaction to a given topic is positive or negative. This is typically used to gauge reactions to products, services, or news topics. Given the emotional nature of this, Sentiment Analysis most closely relates to the challenge at hand. Word polarity is a common and often effective technique. [5]

2.3.2 Statistical Analysis

There exist a number of formulas for gauging a piece of prose. However, these are oriented almost exclusively towards reading level. Two common metrics are the Flesch-Kincaid Reading Ease formula, and the Gunning fog reading level.

$$\mathbf{206.835 - 1.015}\left(\frac{\textit{total words}}{\textit{total sentences}}\right) - \mathbf{84.6}\left(\frac{\textit{total syllables}}{\textit{total words}}\right)$$

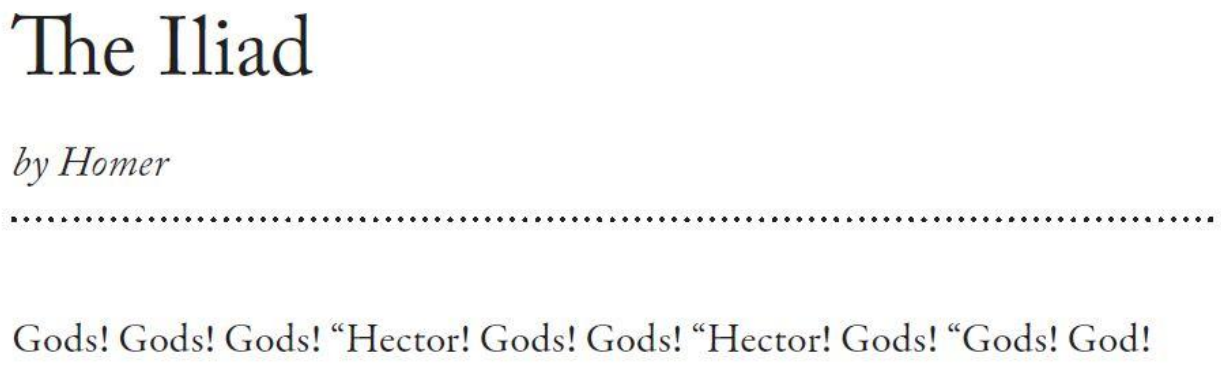
Flesch-Kincaid Reading Ease

$$0.4\left(\left(\frac{\textbf{words}}{\textbf{sentence}}\right)+\ 100\left(\frac{\textbf{complex words}}{\textbf{words}}\right)\right)$$

Gunning fog

2.3.3 Applications

A number of programs currently exist which seek to extend the field of text analytics into overlap with creative writing / literary analysis. Computational linguistics and natural language processing cannot be fully automated as easily, due to the wide variety of possible forms that narrative prose can take. Similarly, automatic summarization is often found lacking when applied to works of literature.



Ten-Sentence Auto-Summarization of *The Iliad* using Word 2008, by Jason Huff

However, programs more properly suited to the task at hand exist today.

2.3.4 Linguistic Inquiry and Word Count

The Linguistic Inquiry and Word Count program, (LIWC / LIWC 2007) is a piece of retail software dealing with the statistical analysis of word usage in documents. The program, designed by James W. Pennebaker, Roger J. Booth, and Martha E. Francis, assesses text documents along a variety of axes, using the principle of a categorical word count.

These documents, in the form of raw text or ASCII in the form of a word processing software package, are parsed to form a selection of 'target words.' These target words are processed through the LIWC framework and compared with a special dictionary constructed for the purpose. Using this, it categorizes each word, fitting it into one or more of approximately eighty categories. A report is then generated, listing total number and percentages of target words which fit into which categories.[1]

LIWC is purely a tool for text analytics, lacking any effort to analyze sentence structure beyond length, pronoun or dialogue attribution, or any real examination of context. Nevertheless, it is powerful due to the large number of categories it provides.

It analyzes a text along lines such as sentence and word length, allowing for an easy examination of reading level. Its dictionary categorizes along metrics of emotional meanings and connotations, and others which help examine the general meaning of, or even the thematic basis of the document. Associations with categories like death, motion, money, the home, etc, can provide a valuable insight into the tone and atmosphere of a work.[2]

Additionally, LIWC examines target words for tenses and common pronouns, the latter categorized into first, second, or third-person. While narrative prose will inevitably contain a mix of these things, the prevalence of one form over another means that LIWC can be used to

automatically assess whether a document was written in first or third-person view, and in past, present, or future tense.

LIWC's dictionary was generated manually. A selection of common words was assembled, and a team of three expert judges was tasked to examine each and categorize them based on the eighty categories mentioned above. Connotations of death, the home, and so on, were impossible for an automated system to examine, mandating the use of a manual examination of thousands of words. However, now that these links have been established, an automated examination, along these lines, of a particular document can be performed in a fraction of a second.

This provides a simple and easy to understand, yet robust tool for quickly analyzing text along multiple vectors.

2.3.5 Text Analyzer

Text Analyzer, by WriteItPro, current version 2.1, is a piece of retail software combining elements of a word processor, with limited automated analysis. It can generate statistics regarding sentence length, and scores along the Flesch-Kincaid and Gunning Fog metrics. However, it does not offer analysis of mood, tone, or the use of other less standardized metrics.

3. Project Description

3.1 Program Overview

LAT must integrate into the existing process of creative writing described in Section 2. The goal is not to alter the writing cycle, but to accelerate it. As such, it must be easy to use, and produce feedback in a straightforward way.

In an ideal world, this would be closely integrated with the word processor used by the writer. This would impact overall ease of use. There are a multitude of word processing programs available, and different writers may be comfortable in different settings. As such, the most universal and accessible platform for LAT is the web app. In a modern setting, any writer can be relied on to have access to the internet. Moreover, if the tool can be run purely client-side, the writer need not actually be connected at any particular moment in order to make use of it. Merely keeping the page in their browser, in a cache, or as a file on their computer will suffice. This is useful to writers who prefer electronic isolation during their writing process.

To implement LAT, JavaScript will be used to create a web app which is straightforward, easy to use, and as self-contained as possible, so it can be easily loaded into a browser and used without an internet connection, if this is desired.

At this stage, the program is defined in terms of core and optional functionality.

Core functions are defined as those necessary for basic program behavior. These must be successfully implemented in order for the project to be completed. It also includes those functions which are not strictly mandatory for this, but which must be implemented to some extent in order for the second phase of the project to be conducted at all.

Optional features will be implemented if it proves both possible and desirable to do so according to exploratory experimentation. Ideally, all optional functionality will be implemented, but in practice, components will inevitably prove unfeasible or merely unimportant to the goal of running the experimentation phase of the project. Nevertheless, serious effort should go into implementing enough optional features to conduct a well rounded experiment and, more importantly, to eventually deliver a product useful to assisting the writing process.

3.2 Core functionality

The LAT is designed to provide immediate, consistent, reliable feedback for a writer, in the form of emotional analysis via LIWC, combined with other useful metrics. This provides the user access to sanity-checks for creative writing, comparable to the error checking of a programmer's integrated development environment. If the writer's intentions for their work are not reflected in the text they are producing, they will have a fast and easy way to check.

3.2.1 Mood/Tone Analysis – Full Text

LIWC is a useful and powerful tool for assessing the emotional and even thematic payload of a given text's language. Its dictionary, categorizing words along such lines, is easily accessible, separate from the program as a whole. LAT will take this library and use it to emulate LIWC's functionality, integrated into the context of a word processor.

This will allow users to perform a rapid mood/tone analysis on their writing, producing immediate feedback, however limited, regarding the emotional content of their work.

3.2.2 Mood/Tone Analysis – Section

Highlighting a section of text should alter LAT's context-menu actions. In this case, its mood/tone word count analysis will be performed only on the appropriate selection of words. This increased granularity will help a writer to scrutinize their work a part at a time.

3.2.3 Mood/Tone Analysis – Change over Time

Automated Mood/Tone analysis of a work by parts will allow the implementation of an automated analysis of Mood/Tone changes through the course of the text. This will be accomplished by creating a moving 'window' of text, approximately one page in breadth. By analyzing the words in this window as it traverses the text, information can be derived regarding changes in Mood /Tone throughout the course of the work.

3.2.4 Indirect Language Analysis

Indirect language, in this context, is that which distances the narrator from his or her statements. A writer may choose these to pad their language, or to soften the impact of their text. However, direct and forceful language is typically preferable. The LAT should be designed to detect examples of indirect filler.

This can be accomplished by creating a database of symptomatic language, and checking for its presence in the text. Examples of such include "generally," "in general," "really," and "very."

As Mark Twain famously advised: "Substitute 'damn' every time you're inclined to write 'very;' your editor will delete it and the writing will be just as it should be."

3.2.5 Word Repetition Analysis

Word repetition represents a common pitfall for an amateur writer. Elegant narrative prose should avoid this, except where it is a deliberate, limited stylistic decision. A simple analysis of word frequency can be useful to a writer, allowing them to quickly evaluate the breadth of their diction.

Detecting the same word used in quick succession in a text would be a relatively straightforward function with obvious benefit. However, detecting which words are viable candidates can prove difficult. The program should avoid highlighting unavoidable word repetition, striking a balance between false positives (pronouns, conjunctions) and ineffectiveness (failing to notice jarring overuse of words across multiple sentences). This should be done in such a way as to be consistent with common writing practices, which themselves must be researched and tested against.

LIWC's dictionary includes categorization for over four hundred function words, which should reduce false positives if implemented correctly.

3.2.6 Punctuation Frequency

Punctuation is an important and frequently overlooked challenge in creative writing, which despite accepted rules of English grammar is often more a matter of style than correctness. Consider the oxford comma. However, a preponderance of certain punctuation marks, such as commas or semicolons, might be indicative of a more serious concern; this metric should help the user to identify such issues and consciously determine their desirability.

3.2.7 Paragraph Length

Sentence length is difficult to calculate, due to multiple uses for the period. However, calculating paragraph length should be quite straightforward, thanks to paragraph breaks and tab indentation.

Paragraph length is important in the crafting of good, readable text. Average paragraph length can easily become disjointedly short, or long to disrupt the reading process.

3.2.8 Word Frequency Analysis with Proximity

Word frequency in a work and proximate word frequency are separate concerns, which can easily appear independent of one another. In addition to calculating total word repetitions, a technique should be implemented to calculate word frequency in a given proximity. For each valid (non function) word, a window should be examined to determine if repetitions occur within close proximity. Exploratory experimentation is necessary to determine the appropriate window size.

Word proximity should be a useful metric regardless of whether the LAT identifies sentence separation. Measuring proximity by word or even character count should be sufficient.

3.3 Optional Functionality

This section details Optional Functionality, peripheral to the core of the project, but to be attempted as proves feasible.

3.3.1 Sentence Length

Detecting the beginning and end of a sentence is a useful tool for automated writing analysis. It is necessary for accurate determination of sentence length, and is the basis for other

forms of writing analysis, including most other features of optional functionality for the LAT project.

3.3.2 Sentence Opening Repetitiousness

With function 3.3.1 completed, it will become possible to analyze word choice in sentence openings. Using the same few words to open consecutive sentences is a common and easily overlooked pitfall in writing. With automatic sentence detection, it should be relatively simple to implement a tool to compile and display statistics on word choice.

3.3.3 Word Tense

Among its other features, LIWC's dictionary includes keywords to indicate past, present, or future tensed verbs. While imperfect, this should help detect sudden, unintended shifts in writing tense. This is a common pitfall for even experienced writers.

3.3.4 Passive Voice

A commonly accepted rule of thumb is to write in active rather than passive voice. True assessment of active voice, and its suitability, is likely AI-complete, but a reasonable facsimile can be achieved simply by finding strings in the pattern of "was <verb past tense>."

3.3.5 Metaphor Identification

Metaphors and similes are important in writing. They represent a means for a writer to bypass limitations posed by language, and so describe concepts that might otherwise be beyond one's ability to describe. Metaphors also allow a means of conveying emotional intensity.[7]

The art of detecting and evaluating metaphors is extensive and challenging. A promising method exists in the form of the Master Metaphor List, a painstakingly detailed compilation of metaphors and metaphorical structures.[8] If this can be implemented reasonably, it could offer

additional insight into the intricacies of writing style without requiring true sentence parsing or AI-complete tasks.

3.3.6 Character Proximity Assessment

By prompting the user to enter the names and pseudonyms for various characters, the LAT could serve several additional functions. Firstly, it could gain some simple insight into the relationships between characters, by measuring recurrence of these names in proximity to one another, and by cross referencing these proximate occurrences with moods as assessed by the LIWC dictionary. In this way, a simple technique could be created to estimate whether characters have a positive or negative relationship with one another.

Secondarily, prompting a user to enter information for their characters, and all ways the character is referenced, will in itself help their writing, by forcing them to think about their characters in this way, when they were introduced, what circumstances introduced them, and what sorts of monikers they give their characters, encouraging greater thought about how to craft and assign these designations.

3.4 Scope Concerns

Unavoidable factors apply limitations to the project, no matter how skillfully executed. It was originally desired to have LAT identify Mood, separate from Tone, separate from Theme present in the work. After more research, this distinction proved to be AI-complete. Only with difficulty, and potentially some guesswork, is a human traditionally able to distinguish between Tone, a writer and character's 'voice' in the story, and Mood, the atmosphere and emotionality of the work as read by a reader.

Identifying Theme, likewise, is something only humans can do, and only with some difficulty and disagreement. No word count or linear regression analysis, no matter how sophisticated, will discern the thematic underpinnings of a piece of literature. The process to do this is most likely involves all the processes of human intelligence, to fully parse, interpret, understand, and contextualize a work in its entirety.

Context-sensitive suggestions, a prose writing autocorrect, are essentially impossible, as this would require the computer to construct prose as an author would. It must likewise be recognized, at every stage of coding and analysis, that all these forms of analysis are suggestions, not an authoritative assessment. Each and every guideline discussed in this section can be broken at the writer's whim. Therefore, stylistically, all feedback must be presented in such a way that acknowledges that the computer may be completely wrong. The concept of 'corrections' is inappropriate in this context, as it could mislead and damage a writer's confidence both in the software and their own writing process.

4. Questions / Hypotheses

4.1 Questions

4.1.1 Is LIWC Emulation for Mood/Tone Analysis Useful to a Writer?

As the core of LAT, LIWC emulation is the first and foremost specific question to be tested. The only useful way to do this is to simply ask users, after they've had the opportunity to experiment and practice their writing while under the purview of the tool.

4.1.2 Are the Auxiliary Metrics Useful to a Writer?

As an extension of the above, the remaining, related metrics of automated creative writing assessment should be judged in much the same way. However, since these are more haphazard and exploratory in nature, a robust response may not be expectable. Therefore, it is important to separate this sentiment from the prior question.

4.1.3 Is the Program Useable in Practice?

However useful LAT's features may be in an abstract setting, the tool is ultimately an application designed to help writers write. If it cannot perform in this task, its other merits, or lack thereof, become much less important.

4.2 Hypotheses

In short, the hypothesis being pursued is to determine whether the LAT, as planned and as implemented, is useful to a writer. This goal, being highly exploratory in nature, can only be assessed by putting it in the hands of writers and allowing them to test it out in a semi-controlled manner.

If their assessment is positive, even partially and with misgivings, the hypothesis can be considered confirmed. At that point, we will have in our hands, a form of automated text analysis which can meaningfully assist in the creative writing process.

5. Project Plan

The project is divided into two parts. First, the program implementation phase: referred to here as the Language Analysis Tool (LAT), and second, the experimentation phase: the creation, collection, and analysis of information relating to the usage of this program by a group of participants. These participants assessed the program's utility along various criteria, described in section 6.

5.1 Potential Pitfalls

LIWC's dictionary contains 4,487 unique words. Individual categories contain only a portion of this. These words are common, and should represent a sizeable portion of each piece of text, in accordance with Zipf's law, which states that utterances of a word is inversely proportional to its rank, such that the most frequent word is twice as common as the second most frequent, and so on.[10] However, in practice, it may prove that only small portions of each text trigger each category frequently enough to produce a meaningful signal visible through the noise.

It may be that a web app of this design is inherently suboptimal for the purposes being pursued. Writers seldom find benefit in moving away from the page being written. It is possible that this will increase distraction and procrastination without corresponding benefit. A tool which exists in-line with the writing process, an extension of a word processor, might be better, although this would carry other disadvantages.

Creating functionality which is both useful and intuitive is challenging, and success is not always easy to determine.

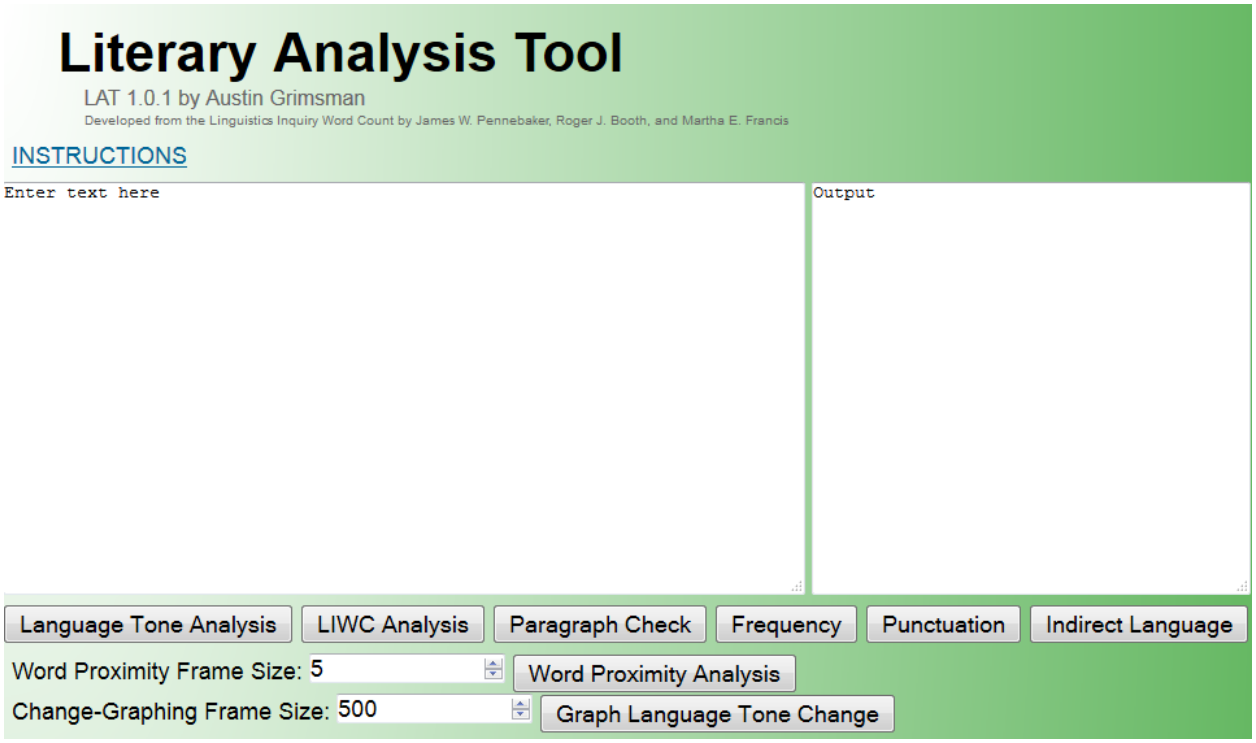
Even assuming optimal functionality, there are a number of concerns present in the endeavor which may prove insurmountable. Analysis is only half an analyst's job. The other half is presenting a case. Even if software could perform objectively useful, flawless analysis, this would not necessarily convince a writer that changes need to be made. Analysis by software carries an air of legitimacy, but also unreliability.

Ideally, the fact that automated literary analysis tools such as LAT are consistent, objective, and impartial should assist in this regard. These tools will present facts uncolored by opinion or agenda. Although a writer might dismiss them as incorrect, there is no element of personal disagreement or bias.

6. Results

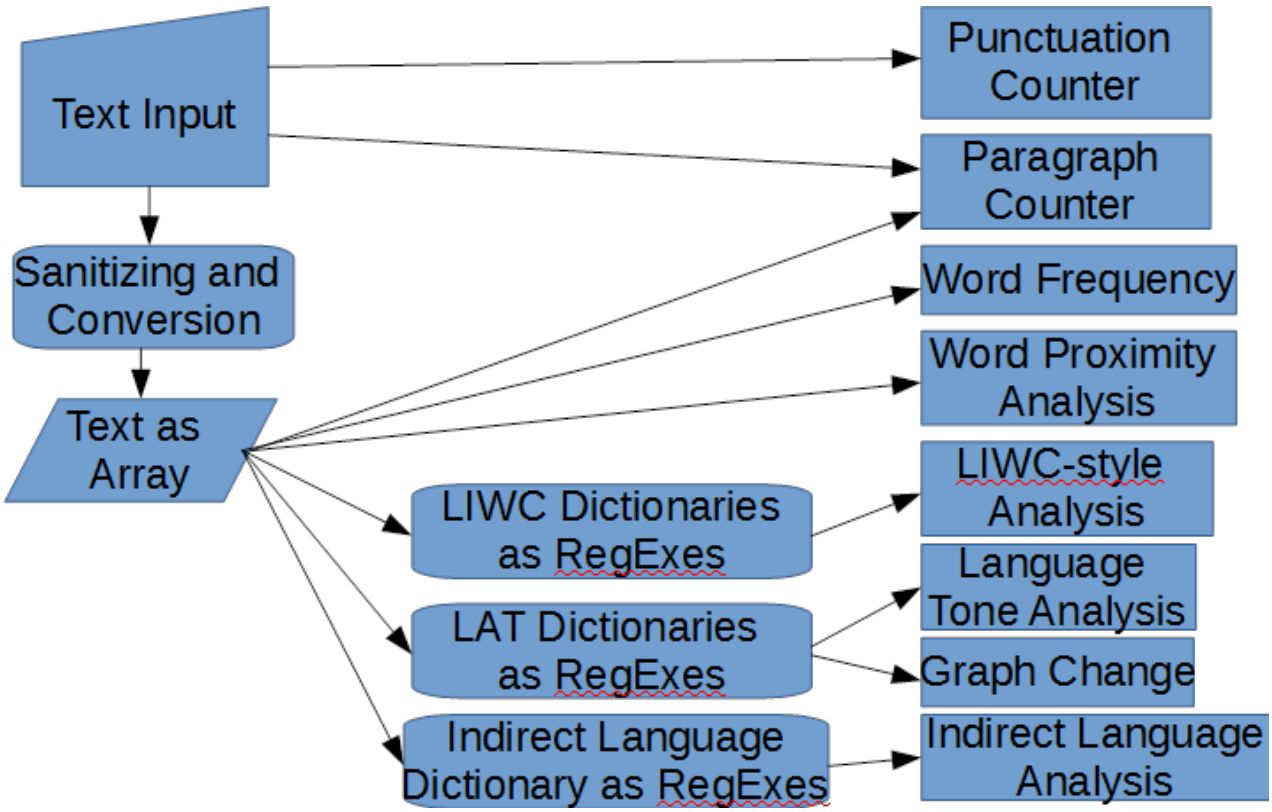
6.1 Functionality of Literary Analysis Tool 1.0.1

The Literary Analysis Tool is a single-page JavaScript web app, currently hosted at www.patterntester.net/LAT. Text to be analyzed is entered into the appropriate text box, and functionality is provided via a series of buttons, producing text and chart-based output.



Additional inputs consist of the sizes of the frames, measured in words, of the word proximity feature and language tone change graphing.

Text input is used to generate results using the following programmatic structure:

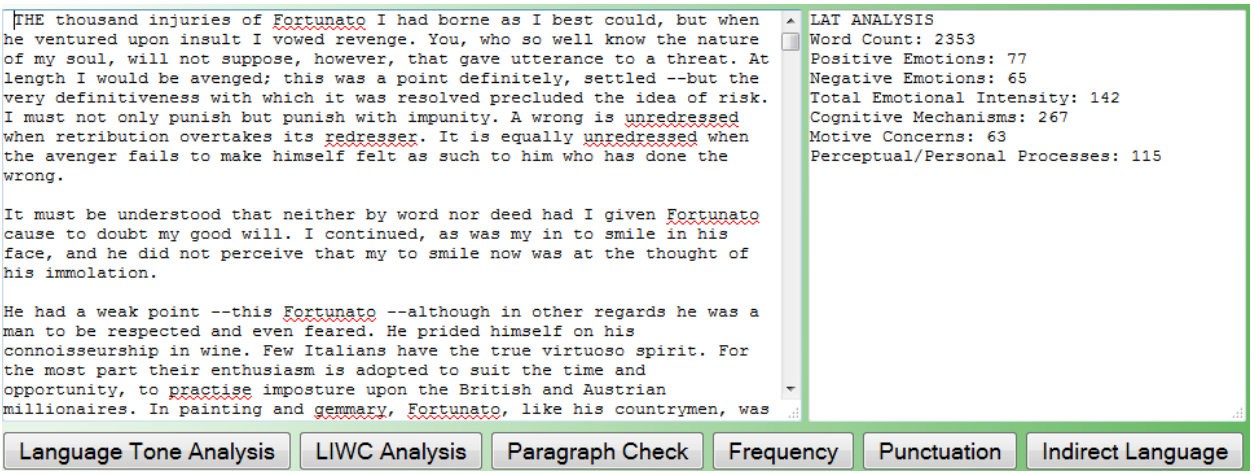


Data sanitation strips input text of all characters other than letters and numbers, which are converted to lower case for string matching. For this reason, punctuation and paragraph analysis requires unsanitized data. Following sanitation, input text is converted to an array of strings, each one word in length.

Most functions require the use of string matching via regular expressions. In JavaScript, regular expressions are stored as their own unique data type. LAT utilizes 68 dictionary files, all stored in this manner. Storage of dictionaries as regular expressions increases efficiency when compared to runtime conversion, or string-to-string comparisons. Executing RegExes on strings, in sequence, produces most of LAT's functionality.

A demonstration of each feature is provided below. Input text is 'The Cask of Amontillado' by Edgar Allen Poe:

6.1.1 Language Tone Analysis



This is the implementation of what has been called "Mood/Tone analysis" elsewhere. This parses input text into an array of strings, and compares every element in this array with every element in each dictionary, keeping a running total of matches. Matches with the PosEmo.js dictionary are displayed as 'Positive Emotions,' NegEmo.js as 'Negative Emotions,' and so on.

In pseudocode:

SANITIZE and PARSE text input into StringArray x

Output x.length as Word Count

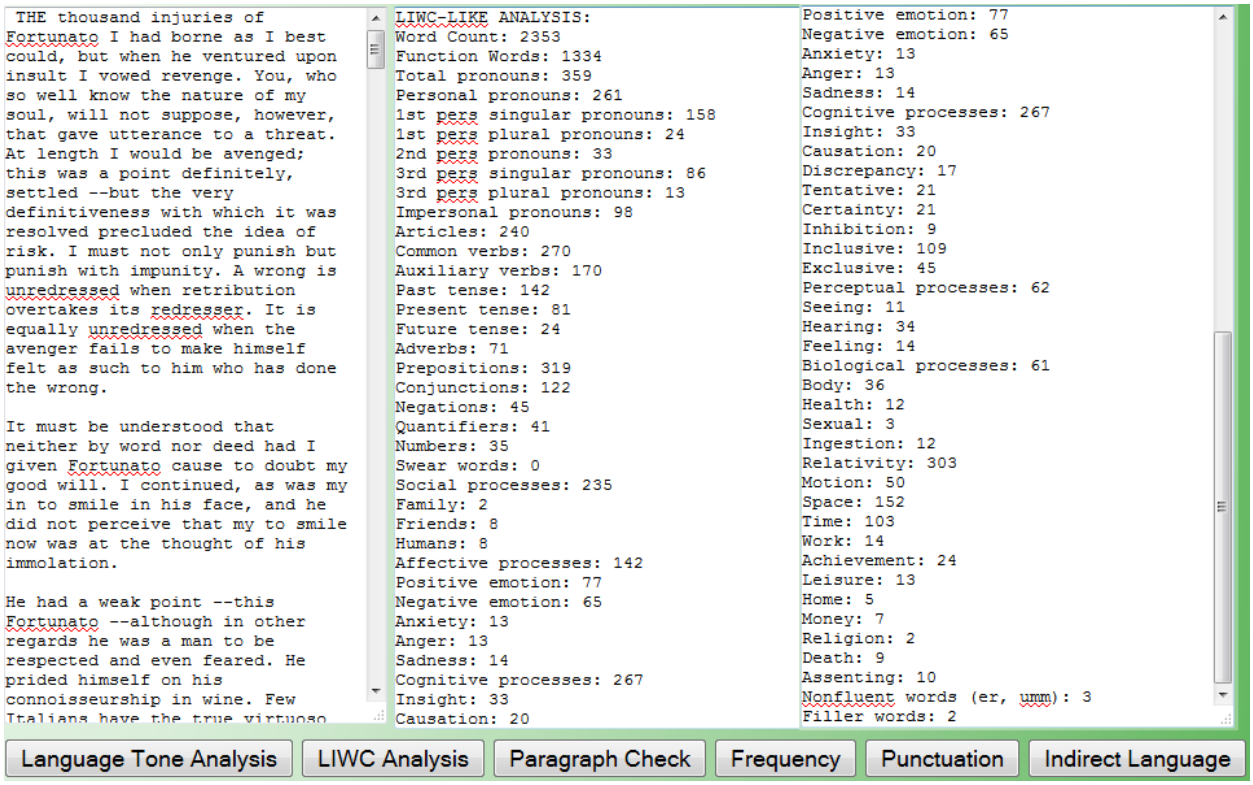
for each element in x

Compare x.element with **all** elements of Dictionaries: PosEmo, NegEmo, CogMech, Motive, PerceptPers

Display total of all matches

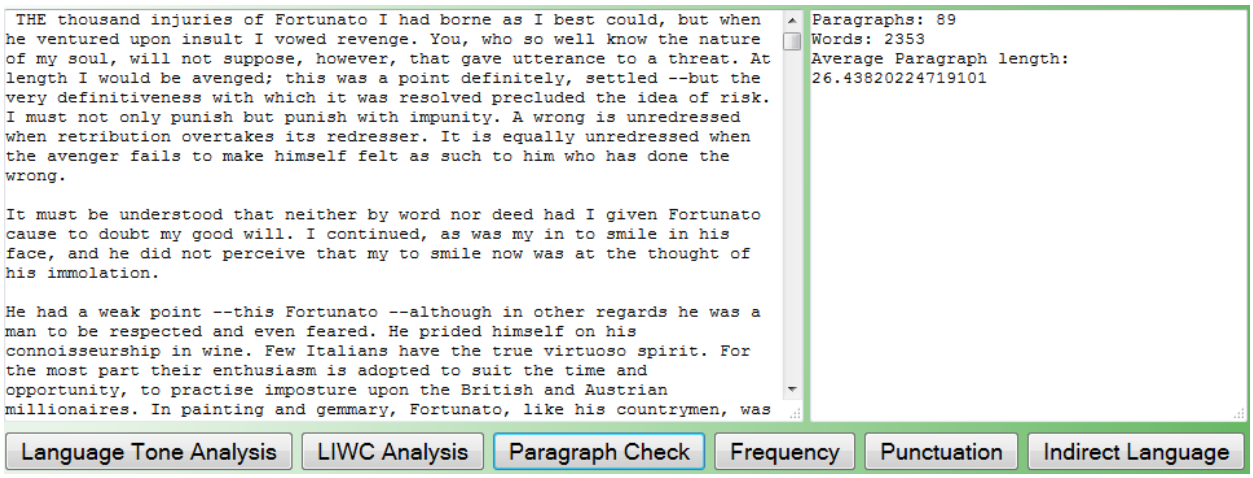
Display combined PosEmo and NegEmo as 'Total Emotional Intensity'

6.1.2 LIWC-style Analysis



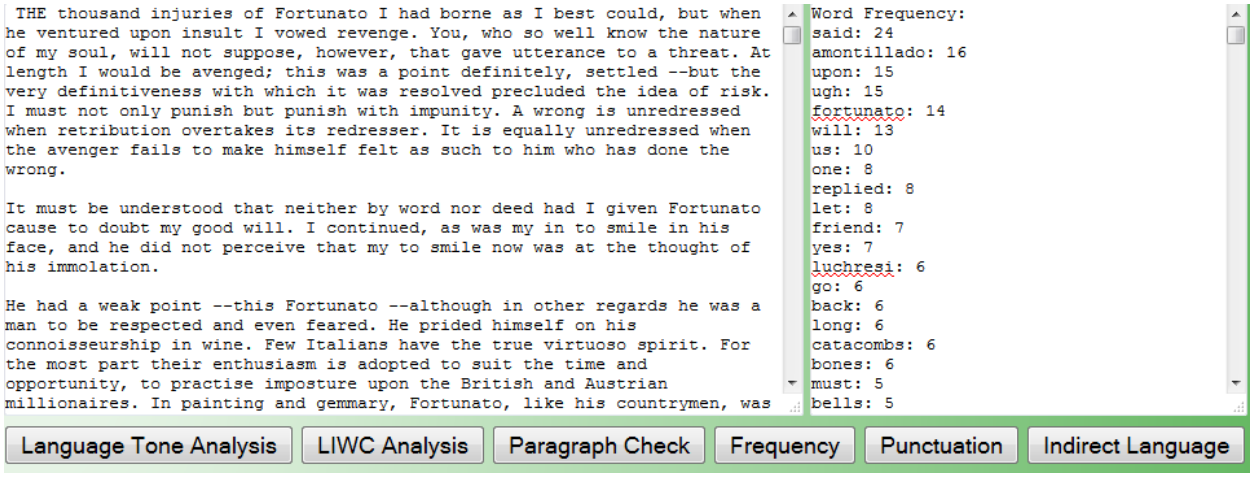
This emulates the function of LIWC for comparative purposes. Its code is structurally similar to Language Tone Analysis above, but instead of five (effectively six) dictionaries, it checks 64, with output displayed accordingly.

6.1.3 Paragraph Check



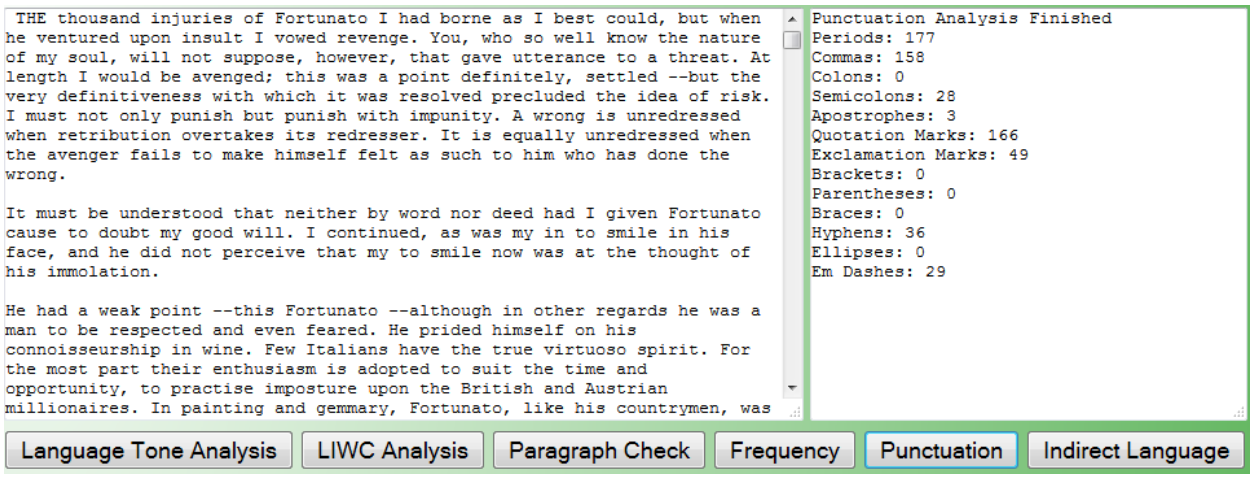
This function scans the input for paragraph breaks between blocks of text and totals them up. Then, it counts words. Then, dividing one by the other, it displays average paragraph length.

6.1.4 Word Frequency Analysis



Beginning with text input sanitized and parsed into an array of strings, this function scans and counts each unique (case insensitive) string. It sorts these by frequency and displays all terms in descending order.

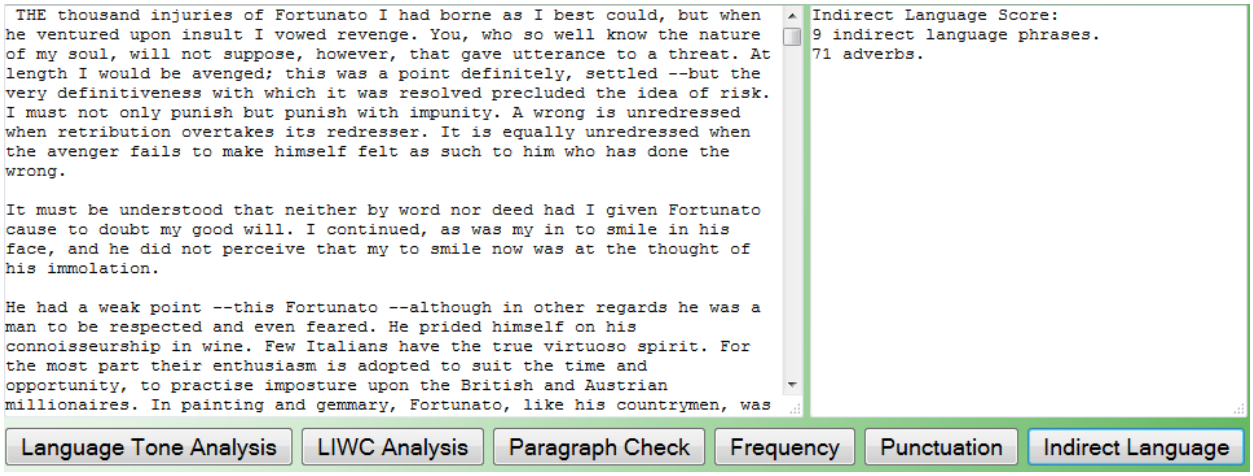
6.1.5 Punctuation Count



This function takes unsanitized text inputs and counts every instance of common (and uncommon) punctuation marks, displaying a list, by which a reader may assess whether they are overrelying on certain forms of punctuation enough to distract.

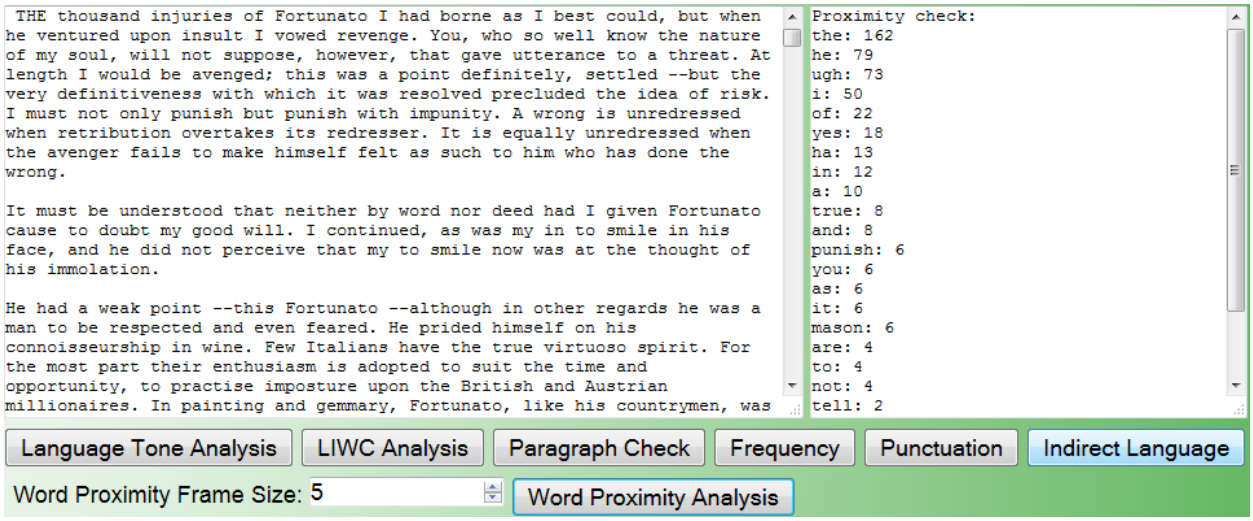
Em Dashes ("--") and Ellipses ("...") are counted first and separately, so that they are not also counted as two hyphens or three periods, respectively.

6.1.6 Indirect Language Analysis



This function begins with a parsed and sanitized array of text. With code similar to Language Tone Analysis, it totals instances of adverbs (the overuse of which may or may not be of concern to the author) and instances of phrases from the indirect language phrase dictionary. For this, sequences of words must be detected, rather than single words only.

6.1.7 Word Proximity Analysis



This function begins with text input converted to a sanitized and parsed array of case-insensitive strings. A frame, or moving window, of user defined length is cycled through the text, detecting any instance of repeated words.

In pseudocode:

SANITIZE and PARSE text input into StringArray x

Read 'Frame Size' as y

for each x.element,

iterate forwards y elements

for each pair of words inside this frame which are identical

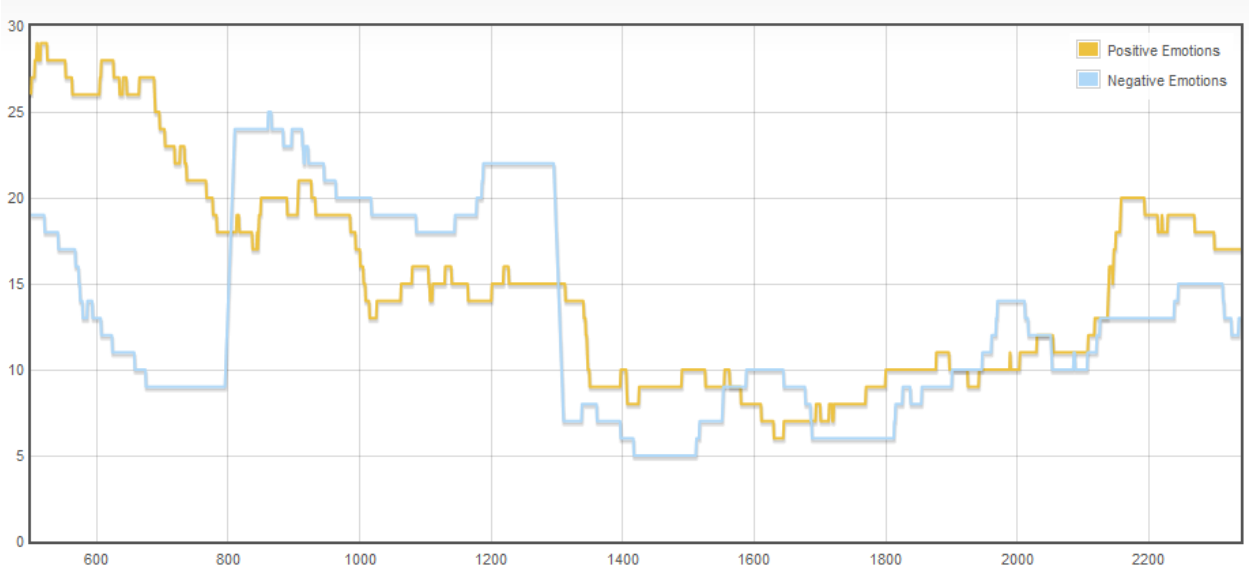
increment score associated with that unique word by 1

OUTPUT list of words in descending order of score

It is possible for the same pair of words to be detected multiple times. The closer the word is to itself, the more often the pair will fall inside the moving window. Therefore, words are

rated both for proximity to themselves, and for recurring repetition. This is intentional, since words repeated in close sequence will be more obvious and distracting.

6.1.8 Graphing Language Tone Change



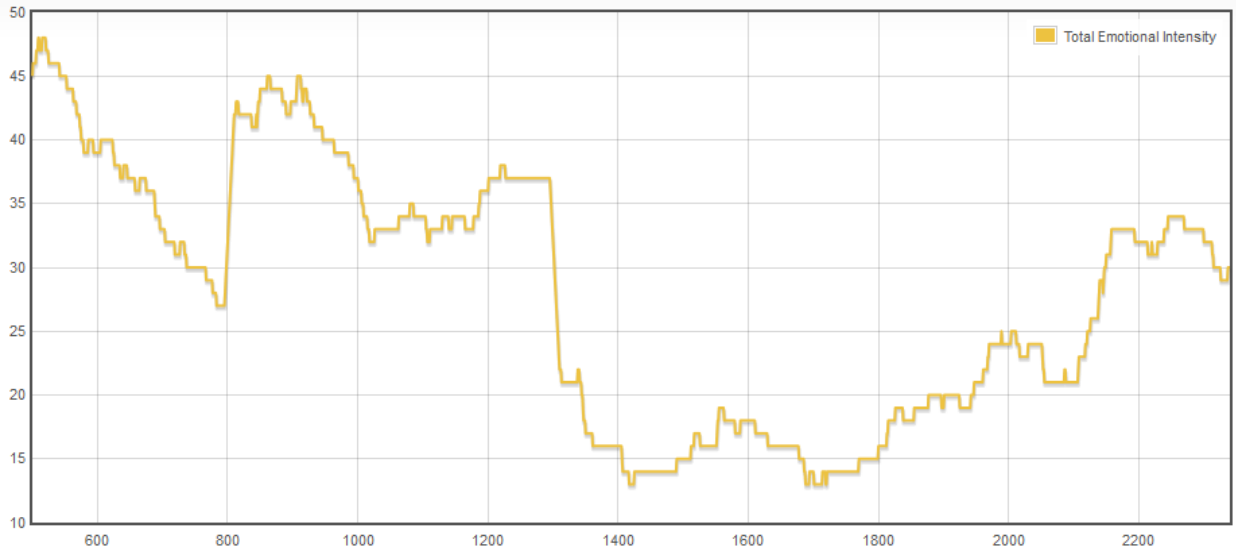
Positive vs Negative emotional intensity graph of 'The Cask of Amontillado' by Edgar Allen Poe. (Frame Size: 500)

This function takes a moving window approach to the Language Tone Analysis, described previously. The user inputs a desired frame size, measured in words. A frame is drawn of that length at the beginning of the text. The Language Tone Analysis function is applied to this window, as if it were its own text. A score is recorded. The frame is then iterated, word by word. As words are matched with the appropriate dictionary, the running total increases. As matching words fall out of the frame, the score decriments, producing data sufficient to form a line graph, as shown in the example above. This is then displayed using the flot.js plugin.

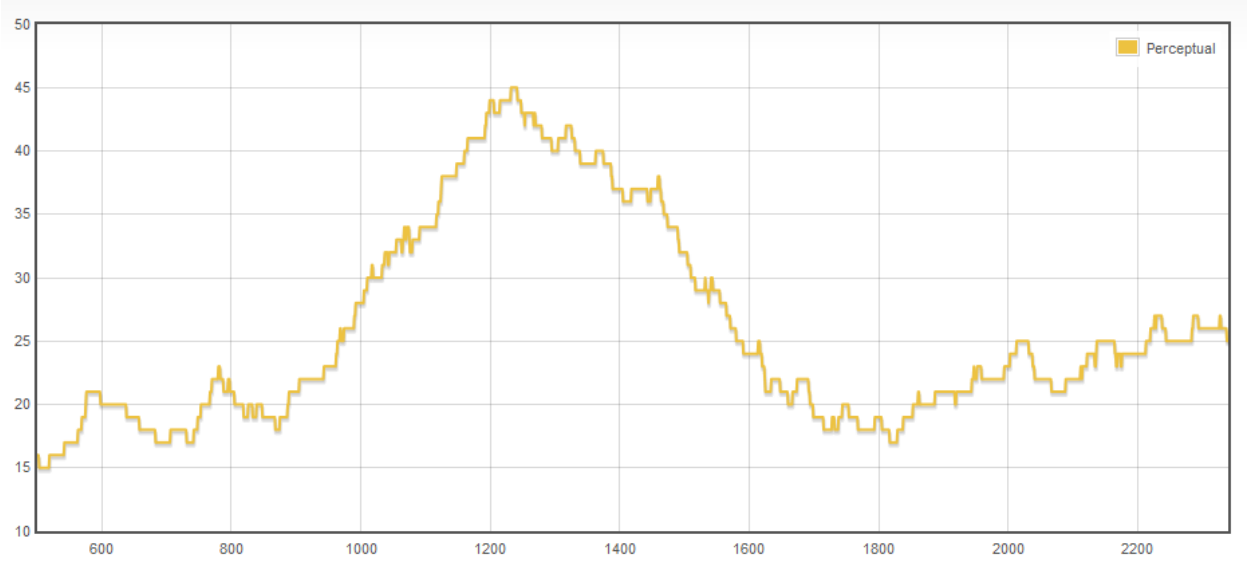
In the above example,"[t]he thousand injuries of Fortunato" fail to create a strong prevalence of negative emotions in LAT. Negative emotions only briefly exceed positive. This

result proved surprisingly consistent. Only the most morose of stories tend to display more negative emotions than positive. This is in spite of the Negative Emotion dictionary being significantly longer than the Positive Emotion dictionary.

Also of note is the shape of the curve: more emotional description during exposition, fewer emotional words during action, and then a second rise in emotion at the story's climax, suggests there is still meaningful, if incomplete, correlation.



Aggregated Emotional intensity graph of 'The Cask of Amontillado' by Edgar Allen Poe.



Frequency of Perceptual words in 'The Cask of Amontillado' by Edgar Allen Poe.

Here, a strong peak is apparent, building up as the characters move into the crypt, where the story's moody atmosphere is most firmly established.

The finished Literary Analysis Tool was uploaded and demoed among a small test audience. This audience consisted of creative writers, ranging from beginners to professional authors. Included with LAT were instructions directing the testers to experiment with the software to their satisfaction, using both preexisting narrative prose they were familiar with, and narrative prose they themselves had written, as well as whatever they felt appropriate in their exploratory use of the software.

6.2 Respondents

The experiment required a relatively difficult-to-find type of respondent, and subject them to a process of experimentation and review which was itself time consuming and effortful. Nine potential candidates were selected, ranging from relatively inexperienced amateur writers, to experienced writers seeking publication, to a published author. These candidates were

approached individually and pitched the basic concept behind the software, and the project. On receiving a positive response, they were provided a link to LAT, along with some basic instructions to get them started.

Of the nine candidates, six satisfactory responses were completed and submitted. It took more than a month for five responses to be returned, with a sixth appearing shortly thereafter. The two-thirds response rate is encouraging, given the time-intensive nature of requested participation, the necessity of filling out a complex survey, and the necessity of learning a new piece of software to evaluate metrics without an unambiguous, preexisting method of 'reading' the output of LAT.

All six respondents were male. Writing experience ranged from one beginner, with less than a year spent learning the craft, to a professional editorial director who has been writing for "as long as [they] can remember."

6.3 Methods

In all cases, candidates were approached via the internet, using direct and indirect personal contacts and membership in online creative writing communities. As such, real names are not always available. In the interests of fairness, names and pseudonyms are hidden, and feedback is referred to by "Respondent #."

Efforts were made to minimize 'priming' of candidates, but keeping interactions to a minimum proved impossible. Simply providing a link to the tool and intervening no further proved problematic, as many respondents did not scrutinize the page closely enough to find the 'Instructions' link. (Limitations in readability and intuitiveness are already clear.) The process of

review required ongoing attention and encouragement to provide complete and honest feedback. In all cases, I made myself available to answer questions as they arose.

Candidates, both in the pitch and the available online instructions, were directed to feed their own stories and preexisting ones into LAT for a period of at least half an hour. They were all given a great deal of time to experiment at their leisure. (Ultimately, it took over a month for five responses to be receipted.)

At the conclusion of their free-form experimentation, they were instructed to fill out an online form survey, producing organized, detailed feedback.

This survey was moderately detailed in nature, with eleven multiple-choice questions and three open-ended survey prompts, allowing potentially paragraph-length responses. Additionally, the survey asked for rudimentary personal information, limited to name, gender, and amount of time spent pursuing the art of creative writing.

The survey used a 1-to-5 Likert scale measuring the perceived utility of the software in nine categories, followed by a 'final verdict' of two yes-or-no questions. The Likert-based questions asked users to judge the utility of LAT's eight key functions, followed by an overall assessment of the worth of the program as a whole.

6.4 Responses

Respondents were given the following nine prompts, and required to select a response on a Likert scale, ranging from 1, "Not Useful," to 5, "Extremely Useful," as illustrated below:

Example of survey input.

Responses were as follows:

Tester/Respondent	#1	#2	#3	#4	#5	#6	Mean
How much experience do you have writing narrative prose?	Six to ten years	One to three years	Three to six years	One to three years	"As long as I can remember"	Less than one year	
How useful did you find the Language Tone Analysis function?	4	3	3	5	3	2	3.33
How useful did you find the LIWC Analysis function?	5	2	4	3	4	1	3.17
How useful did you find the Paragraph Check function?	2	2	3	3	5	2	2.83
How useful did you find the Frequency function?	3	2	2	5	5	3	3.33
How useful did you find the Punctuation function?	2	1	4	3	4	2	2.67
How useful did you find the Indirect Language function?	3	2	3	4	4	2	3.00
How useful did you find the Graph Language Tone Change function?	5	3	1	5	3	3	3.33
How useful did you find the Word Proximity Analysis function?	3	2	2	5	4	4	3.33
Overall, how useful did you find the Literary Analysis Tool?	5	2	3	4	4	2	3.33
Would you seriously consider using the Literary Analysis Tool to assist your creative writing process?	Yes	No	Yes	Yes	Yes	No	
Would you seriously consider using the Literary Analysis Tool to assist your writing process if it were further refined in the future?	Yes	Yes	Yes	Yes	Yes	Yes	
Mean Numerical Score	3.56	2.11	2.78	4.11	4.00	2.33	

Responses indicated that there was perceived value in the current implementation of LAT. Two thirds of responses affirmed that users would "seriously consider using the Literary Analysis Tool to assist [their] creative writing process."

All responses expressed a belief in the value of LAT as a concept, either as implemented or if further refined. However, this should be tempered both by the small number of responses (six) as well as the likelihood that users most interested in the project would be those most inclined to respond.

7. Discussion

7.1 Interpretation

The exploratory nature of this project, as well as the intangibility of writing quality, initially suggested that LAT would be of dubious value to a creative writer. However, following design and implementation, the experiment produced responses more positive than was expected. It was at first believed that the initial implementation would be useful only as a boilerplate proof-of-concept from which more refined future work could be created, which would only later display utility for a budding creative writer. Judging from the preliminary trial experiment, this view has proven incorrect.

Instead, responses consistently suggested a perceived value in LAT 1.0.1; the tool as it presently exists. Two thirds of respondents indicated interest in using the tool as it exists to assist the writing process, and interest in the tool conceptually, with further refinements, were unanimous.

Whether this view reflects the existence of objective utility is not known. There are possible factors at work which suggest caution when interpreting the meaning of these preliminary responses. This is a relatively novel use of text analysis, and this newness might skew results. LAT's graphical representation of creative writing data, a unique feature, produces novelty factor. This may result in confirmation bias among respondents. Additionally, the tool works with an objective analysis to produce scores, which carries with it the suggestion of objective correctness and intrinsic value where this is not necessarily the case. In this view, since the tool produces output automatically calculated using elements of the text, one presumes that it has meaning and that this meaning has value.

Nevertheless, these positive responses are encouraging. If nothing else, it demonstrates that creative writers can accept feedback from a piece of software, and consider it valid. This alone is a significant finding, given the distrust users were expected to feel about a machine second-guessing their work.

These positive verdicts were not without qualifications, and the survey's commentary sections produced some specific and potentially useful complaints.

Respondents expressed some confusion regarding the meaning of absolute scores. Relative scores, meaning how the various scores of a passage changed over time, were useful since they could show change over time, allowing a work to be compared with itself. Scores measured as an absolute, however, were not easily understood. A quantified value did not necessarily result in quantifiable interpretation. Respondents had no context by which to determine the meaning of a 30 in emotional intensity, and what that score meant for the text.

This was partly intentional. The experiment was considered exploratory in nature. Participants were expected to discover for themselves what each score represented through personal experimentation and repetition. It was considered inappropriate to bias respondents by producing any form of template by which to judge a text. Furthermore, this reflected the efforts taken to avoid the appearance of an automatically generated verdict on any inputted passage. LAT was not intended to state conclusively whether any inputted passage is good or bad in any specific way, since efforts to automatically judge writing quality were considered ultimately impossible and the effort counterproductive.

Nevertheless, a lack of any existing structure produced confusion in responses. This effort to avoid passing an automated value judgment merely transferred to the user the task of

deriving judgment from automatically generated scores. This lack of structure removed a basis by which users could feel confidence in the patterns of results they observed.

Simple, numerical feedback metrics, such as sentence length, paragraph length, and punctuation usage, as well as indirect language detection, proved less popular than emotion-based feedback, and word proximity analysis. This is encouraging and perhaps unsurprising. Emotion-based LIWC (and derivative) functions provide the most unique function of the software. Sentence and paragraph length metrics are commonplace, but users indicated a liking for LAT's unique ability to produce objective emotionality metrics, and display these in a relatively intuitive fashion.

The direct LIWC-emulating function proved slightly less popular than LAT's customized Language Tone Analysis. The former dutifully displays all sixty four of LIWC's original metrics, while the latter produces just six key metrics.

The word proximity metric was questioned in its current implementation. Responses suggest that this metric could be improved with a different display. Presently, word proximity is represented by a single score per word, indicating both the frequency and relative proximity that word is used in conjunction with itself. This does not actually display specific instances where a word is repeated, either via contextual highlighting or other means. This suggests a more intuitive and useful means by which this information could be conveyed, enhancing the utility of the feature.

Some skepticism was even expressed in the graphed change-over-time analysis, considered the most novel and, in theory, useful component of LAT as implemented. However, even those respondents expressed belief in the value of future versions of LAT. This suggests a

belief in the value of automated feedback as a technique to enhance or accelerate the writing progress, regardless of the perceived value of any one particular component.

7.2 Benefits

Evaluating works of creative writing is, firstly, of utility to teachers and students of the field. It is a subject of debate whether creative writing is a subject that can be taught, as one might teach mathematics or history. It is an art form, and a skill, more than a science. While this applies challenges to its automatic evaluation, it enhances the utility of those evaluation processes. Any tool which can assess, analyze, and score creative writing is of great utility in this regard, because the field requires a great deal of practice to gain a basic proficiency in writing narrative prose.

As such, any way to score creative writing, however crude, can streamline feedback to the writer, and so streamline the act of practice and evaluation, and is thus of great importance to the training process a creative writer undergoes.

More broadly, any increase in the ease or efficacy of automated text assessment is useful in schooling. In nearly any field, students are expected to write essays, which must be graded. The process of reading and grading large numbers of essays or other word problems is one of the largest obstacles facing the modern professor.

Online classes face the same bottleneck on a more extreme scale. Online universities allow a single professor to teach a very large number of students, limited only by their ability to answer questions and to grade essays. Other forms of grading can be easily automated, but essay writing can only be effectively graded with manpower. For this reason, many students never get the chance to practice writing essays until their final examinations.

While the Literary Analysis Tool will not revolutionize online essay grading, its immediate feedback and truncation of the review cycle should have potential in teaching the art of creative writing.

Quick and consistent feedback can accelerate the process of review and revision, allowing more passes over a piece of text prior to publication. Automation, in the form of LAT, produces this feedback. It cannot replace the insight and understanding of a human reader, but contrariwise, a human reader cannot replace the functionality which can be provided by automated analysis.

Human readers have personal bias, regardless of efforts to remain objective. The process of reviewing a story is slow and effortful. Even when adequate time is available, it is difficult work, frequently avoided if possible. This required effort further reduces consistency of feedback, since minor variations in personal mood, situation, or fatigue level can result in very different responses to the same text. Self-review proves even less consistent, since a reader has a mental image of what their story ought to be, which derives from their intent rather than the text which exists.

The writing process requires cycles of revision. Obtaining human review can add anywhere from hours to months of time to each cycle, assuming it is affordable or available at all. Self-review, if it's to provide any serious benefit, must take place after the writer has put their story out of mind long enough for the initial intent to fade, allowing a less-biased review process to take place. This takes weeks or months.

With LAT, and hypothetical future review tools, the process of review happens in seconds. With the press of a button, meaningful data is provided to a writer. This is derived from

the text which exists, rather than that which was intended, or even that which was subjectively experienced by a test reader. The analysis, though not benefitting from a human thought process, is unbiased and consistent. A set input produces a set output, every time, without situational interference, without confusion, without personal preference.

Results thus far indicate that LAT produces useful feedback, or at least the perception of useful feedback. This should serve to accelerate the process of review and rewriting.

The existing slow, manual process of review means that many authors function with little to no meaningful review at all. Receiving meaningful feedback is difficult enough for projects like novels, which have release schedules measured in years. Today, much fiction is released with shorter cycles. Web original fiction, ranging from internet magazine publication to fiction forums, is typically published a chapter at a time, often in rapid succession. Review cycles measured in weeks are inadequate for the task, and so these forms of fiction frequently go without.

Automated review changes this formula. If a review pass happens in seconds, no time is wasted and no effort or resources are expended to obtain it. It becomes feasible for even free, hobbyist fiction, published a chapter at a time, to undergo multiple cycles of revision.

This accelerated cycle has the potential to change what a review cycle represents to a writer. If review takes place in seconds, the desire to only send out major versions changes as well. For the first time, it becomes feasible to review stories as quickly as they're edited. As a programmer today can quickly revise code thanks to quick compilers, with automatic review software, a writer can submit a text, receive feedback, make a small change, and repeat until completion.

Automated review has the added benefit of avoiding the social effects of human judgment. Submitting an unfinished story to a human, especially one skilled at reviewing, can be an embarrassing, stressful process, which may involve discussion or argument when this is not desired. Submitting a story to a machine, however, carries little risk of the same.

Evaluating works of creative writing is, firstly, of utility to teachers and students of the field. It is a subject of debate whether creative writing is a subject that can be taught, as one might teach mathematics or history. It is an art form, and a skill, more than a science. While this applies challenges to its automatic evaluation, it enhances the utility of those evaluation processes. Any tool which can assess, analyze, and score creative writing is of great utility in this regard, because the field requires a great deal of practice to gain a basic proficiency in writing narrative prose.

As such, any way to score creative writing, however crude, can streamline feedback to the writer, and so streamline the act of practice and evaluation, and is thus of great importance to the training process a creative writer undergoes.

In more general terms, any increase in the ease or efficacy of automated text assessment is useful in schooling. In nearly any field, students are expected to write essays, which must be graded. The process of reading and grading large numbers of essays or other word problems is one of the largest obstacles facing the modern professor.

Online classes face the same bottleneck on a more extreme scale. Online universities and their ilk can allow a single professor to teach a very large number of students, limited only by their ability to answer questions, and to grade essays. Other forms of grading can be easily automated, but essay writing produces manpower requirements which technology, as yet, has not

been successful in reducing. For this reason, many students never get the chance to practice writing essays until the final examination.

While the Literary Analysis Tool will not revolutionize online essay grading, its immediate feedback and truncation of the review cycle should have potential in teaching the art of creative writing.

7.3 Future Work

Word Proximity Analysis would greatly benefit from a complete redesign. Instead of producing a numerical score for each word, it should resemble a spell check, highlighting repetitions of words within a certain window. This would allow users to immediately pinpoint those repetitions in context, to determine if this has a deleterious impact on the flow of their text.

Sentence detection could prove useful. This would allow for the implementation of reading level calculation, enhance current functions such as paragraph calculation, and open the door for future endeavors, such as examining variability of sentence length. Alternating intermittently between long and short sentences is considered useful to maintain reader interest. A long string of sentences of equal length can serve to create a plodding tone. Alerting a writer to this could prove beneficial.

LAT is currently limited to derivatives of LIWC's dictionaries. Expanded or alternative dictionaries could help increase the signal to noise ratio of the Language Tone Analysis function. Pairing the base LIWC-derived dictionaries with a thesaurus could accomplish this, in theory. However, synonyms often have different connotations. “Greedy” and “hungry” are synonyms, yet carry radically different subtexts and emotional weights. As such, building an expanded

dictionary is likely to prove a daunting and time consuming task. Other dictionaries exist, and these should be investigated through future incarnations of LAT.

Pairing some LAT functionality with a word processor could prove beneficial, in order to avoid the distractions inherent in connecting to the internet, and the breaking of flow. If automated feedback is useful, minimizing the time and effort to generate that feedback increases its utility. Adding LAT-like functionality as an extension to software could achieve this. This benefit might be marginal compared to the necessary workload and loss of universality inherent in a web app. Better still might be to produce variants of LAT, with different dictionaries and different platforms, oriented to different goals. Introductory versions could be designed to teach creative writing skills. Another could be designed with a seasoned writer in mind, deemphasizing tonal detection while emphasizing features like word repetition and sentence length variation. A plug-in could be integrated into web publishing software to assist with blogging.

While demoing LAT, a bias towards detecting positive emotions over negative emotions was noted by test respondents. In all but the most morose stories, positive emotions consistently outscore negative emotions. This is in spite of the negative emotion dictionary being larger. (498 negative terms vs 408 positive terms) The cause of this is presently unknown, and may merit further study.

Establishing benchmarks would assist future versions of LAT. Respondents did not know which scores represented what. By analyzing many corpora, baselines could be established, contextualizing scores generated by tonal analyses.

7.4 Retrospective

This project proved a challenge more in planning than in implementation. The actual programming was not especially groundbreaking or difficult. However, determining what to program was constantly in flux. The project saw significant changes in capability between the initial proposals and the final chosen design, as evidenced by the discrepancies between the proposed capabilities in Section 4 and the final product described in Section 7. For the sake of describing the project's evolution, the earlier section has been left very similar to the way it was first written.

With the benefit of hindsight, I would have more aggressively pursued a final design document. The project evolved greatly over time, from what was initially conceived of as a custom word processor, to a Word plug-in to an interface between Word and LIWC, and finally to a web-based project utilizing LIWC's dictionaries. If I were to do it over, I'd have avoided wedding the project to LIWC's dictionaries as closely as I did. Other, similar dictionaries are available, and drawing from a wider variety of sources could have produced more interesting results.

8. References

[1] Pennebaker, J. W., Booth, R. J., & Francis, M. E. (2007). Operator's Manual Linguistic Inquiry and Word Count: LIWC2007. *The University of Texas at Austin and The University of Auckland, New Zealand*.

[2] Tausczik, Y. R., & Pennebaker, J. W. (2010). The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of language and social psychology*, *29*(1), 24-54.

[3] Ananiadou, S. (2009). Adding Value to Scholarly Communications & Repositories through Text Mining. Retrieved January 1, 2013.

[4] Clifton, C., Cooley, R., & Rennie, J. (2004). TopCat: data mining for topic identification in a text corpus. *Knowledge and Data Engineering, IEEE Transactions on*, *16*(8), 949-964.

[5] Mishne, G. A. (2007). *Applied text analytics for blogs*.

[6] Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, *2*(1-2), 1-135.

[7] Williams-Whitney, D., Mio, J. S., & Whitney, P. (1992). Metaphor production in creative writing. *Journal of Psycholinguistic Research*, *21*(6), 497-509.

[8] Shutova, E. (2010, July). Models of metaphor in NLP. In *Proceedings of the 48th annual meeting of the association for computational linguistics* (pp. 688-697). Association for Computational Linguistics.

[9] Ramsay, S. (2003). Special Section: Reconceiving Text Analysis Toward an Algorithmic Criticism. *Literary and Linguistic Computing*, *18*(2), 167-174.

[10] Adamic, L.A. (2003) Zipf, Power-laws, and Pareto - a ranking tutorial. Retrieved from http://www.hpl.hp.com/research/idl/papers/ranking/ranking.html

[11] Hespos, S. J., & Spelke, E. S. (2007). Precursors to spatial language: The case of containment. *The categorization of spatial entities in language and cognition*, 233-245.

9. Appendix

9.1 Figures

Figure 9.1.1: Use Case diagram of program usage.

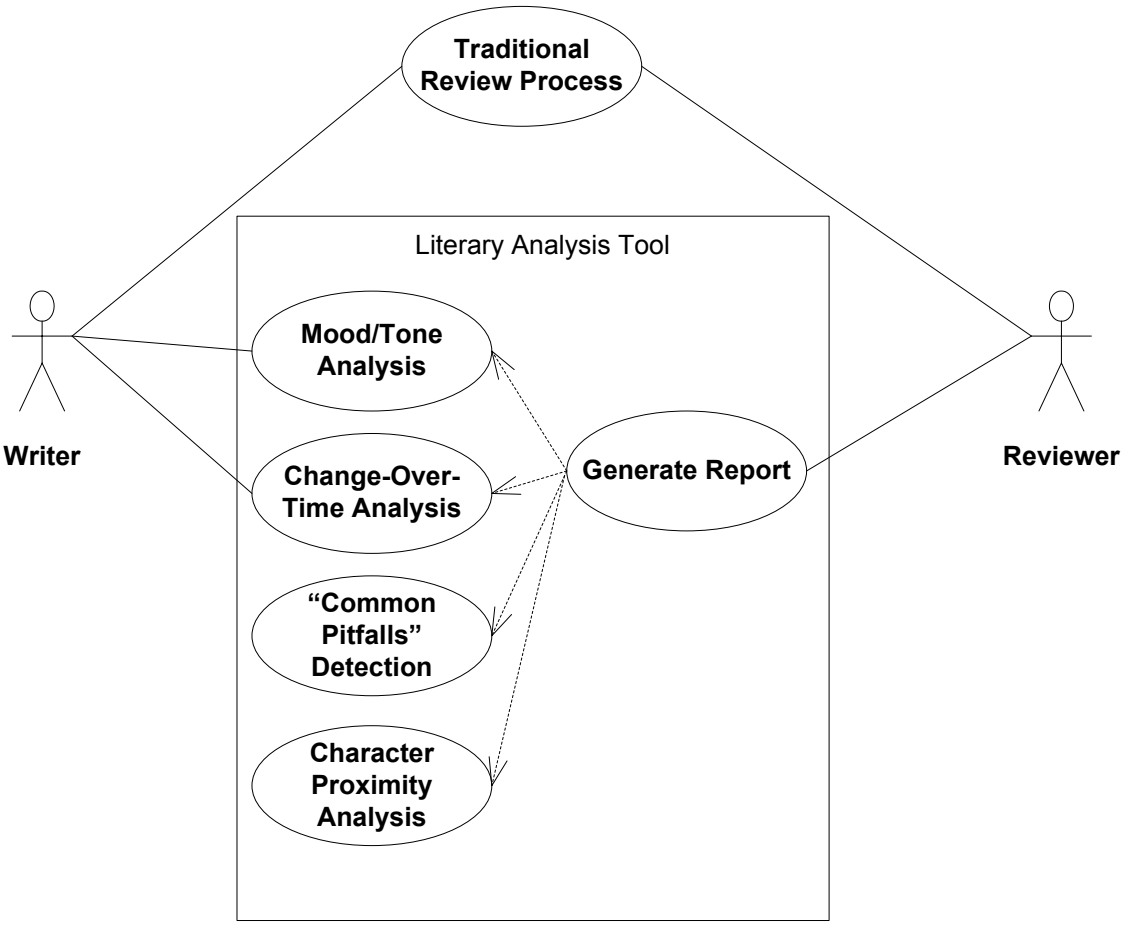


Figure 9.1.2

Use Case Name:	Mood/Tone Analysis
Scenario:	Users create analysis of text's mood/tone
Trigger:	'Analyze Tone' button in UI is pressed
Description:	Primary functionality of program, in emulation of LIWC, iterating through

	text to determine word counts along categories of specific emotional payload
Actors:	Users
Preconditions:	Text has been entered into input area
Postconditions:	Report is generated and displayed onscreen describing the LIWC-derived mood/tone analysis of the text in the input area.
Normal Flow	-User copies text into ctrl-c buffer and pastes in appropriate text field -User selects appropriate button from UI -JavaScript on local machine analyzes text and generates report -Report is displayed in text field
Notes:	This is the basic core functionality of the Literary Analysis Tool, without which no meaningful functionality or purpose can be said to exist.

Figure 9.1.3

Use Case Name:	Change-Over-Time Analysis
Scenario:	Users create analysis of how text's mood/tone changes over time
Trigger:	'Change Over Time' button in UI is pressed
Description:	Derivative function of earlier Tone analysis; examines a moving window of text analysis along such lines, and produces a report of top-level categories
Actors:	Users
Preconditions:	Text more than 1000 words in length has been entered into input area

Postconditions:	Report is generated and displayed onscreen describing how the mood/tone analysis changes over time.
Normal Flow	-A moving window of 500 words is selected. -Inside the window, a Tone analysis is generated -The window is iteratively moved forwards, with more reports generated successively -Alterations in results are analyzed and presented
Notes:	The precise method by which this information is best collected and displayed still requires some experimentation and future development.

Figure 9.1.4

Use Case Name:	"Common Pitfalls" Detection
Scenario:	User checks text against predefined common pitfalls in writing
Trigger:	'Common Pitfall' button in UI is pressed
Description:	A series of separate, simpler analyses are performed
Actors:	Users
Preconditions:	Text has been entered into input area
Postconditions:	A multi-faceted report is generated and displayed in the UI
Normal Flow	Compare the entered text with the following techniques: -Indirect Language Detection

	-Word Repetition Analysis -Punctuation Frequency Analysis -Paragraph Length Assessment - Word Proximity Analysis
Notes:	This feature is subject to the most change, out of core functionality. Certain functions may be later spun off into their own functions with their own UI buttons.

Figure 9.1.5

Use Case Name:	Character Proximity Analysis
Scenario:	Users launch subroutine assessing character name proximity
Trigger:	User selects appropriate option in UI
Description:	Assessment is run where one or two names are cross-referenced throughout the text, and proximate word analysis is performed to characterize the text's attitude towards them and their relationship.
Actors:	Users
Preconditions:	Text has been entered into input area
Postconditions:	A report has been generated and displayed in the second, feedback text area of the UI.

Normal Flow	-User is prompted to type one or two character names -Assessment is run where words are scored based on their proximity to these proper nouns, and in turn, with one another. -Most strong tone designations are returned to the User
Notes:	This feature is subject to large amounts of change and experimentation.

Figure 9.1.6

Use Case Name:	Generate Report
Scenario:	Information from all types of analysis is collected and exported
Trigger:	User selects appropriate option in UI
Description:	A report is exported on behalf of the user.
Actors:	Users
Preconditions:	Text has been entered into input area
Postconditions:	A text file has been outputted containing a report with all information in all other functions.
Normal Flow	-User selects option in UI -ALL other use cases are run -A file is created displaying all information which might otherwise be displayed in UI's output