Pollock Parser Final Write-up

**1. A description of the motivating problem.**

Rather than address a pressing problem in the field of computational linguistics, we chose to focus on the implementation of a unique artistic tool. Taking inspiration from influential 20th century abstract expressionist painter Jackson Pollock, we set out to create a bridge between emotional expression in written word and visual art. Pollock’s style, which utilizes seemingly random splashes of raw color to illustrate abstract themes, offered an approachable model for projecting raw emotion from other forms of media to art. This is mainly due to the ease with which nearly random displays can be produced with computer code. Our end goal was to analyze linguistic corpora, primarily in the form of poems, excerpts, and short paragraphs, and generate a Pollock-style “splatter plot” representing the piece’s prevailing expressed emotion.

**2. A brief survey of related research, citations to the algorithms you implement, and a description of similar existing products.**

There are various examples of visual text-interpreters online, such as Tagul, which takes text as input and generates a word cloud from specified colors, angles, fonts, and other parameters explicitly defined by the user (and are thus constrained to the user’s few preferences). Other programs that dynamically produce artwork (e.g. [Painting Fool](http://www.newscientist.com/blogs/nstv/2012/01/painting-computer-surprises-viewers-with-its-artwork.html)) also already exist. This highly advanced program creates formidable artwork based on web searches the software performs itself. This is fairly different, however, from our word-to-art project idea.

[Wordle](http://www.wordle.net/) and [ImageChef](http://www.imagechef.com/ic/word_mosaic/) are quite possibly the programs most similar to our own. They accept words and create an artistic word shape by coloring, bending, and warping words into some cohesive form. By contrast, these programs still lack is an intrinsic link between the sentiment of words submitted by the user and the actual form created by the program. Case in point, the user can input “Despair” and manually instruct the program to arrange the word bubble into a heart shape and write the words in hot pink and comic sans font. Our project is so far the only one we’ve found that renders the shape and tone of the artwork directly based on the inputted words themselves. Having minimal user input also contributes to an increased sense of impressiveness from using our program.

In terms of algorithms, we built the program from scratch ourselves, but we used two corpora (positive\_words.txt and negative\_words.txt) provided to us in the Sentiment Analysis Resources on Piazza (Hu and Liu KDD-2004). For the files that we used to manually map words to colors and emotions, we used a list of the 1000 most common English words from wiktionary.org (<http://simple.wiktionary.org/wiki/Wiktionary:Most_frequent_1000_words_in_English>). Furthermore, to associate emotions with colors, used a paper entitled “Colorful Language: Measuring Word-Colour Associations” by Saif Mohammad at the Institute for Information Technology in Ontario, Canada.

**3. Feedback from users, if available**

When obtaining user feedback, we asked other students to react in terms of how accurately our color palettes reflect various emotions as well as how effectively the program can represent the emotional connotations of particular texts. As with most art, personal opinion and subjectivity produced multiple ideas on what our artworks represented. Happily, varying opinions varied only within emotions that were similar or “neighbored” the emotion the program intended to model. For example, some viewers may not have immediately recognized the program’s depiction of a love-prevailing poem as depicting love, but the emotions they listed were those with similar representative color palettes (e.g. surprise and joy, love and trust, sadness and fear).

**4. Analysis of any shortcomings of your program, and thoughts on future development**

The major limitation of our project stems from the fact that we had to manually assign emotions and colors to individual words (as we were unable to find appropriate corpora mapping words to such features online). As a result, visual representations corresponded to our personal inclinations towards characterizing language instead of something more widely-accepted. In addition, Pollock’s paintings do contain elements that transcend purely random artistic expression, like the dominant black brushstrokes in his work entitled [*Number 11, 1952*](http://upload.wikimedia.org/wikipedia/en/2/2d/Blue_Poles_(Jackson_Pollock_painting).jpg). We feel that accounting for the randomness alone was within the scope for this project, so identifying dominant textual features and visually representing them would have been suitable for longer-term projects.

Future iterations of this project might take a completely reverse approach from ours, wherein a program might analyze *visual* corpora and produce a *linguistic* analysis of it. This would probably require a good deal of Artificial intelligence and machine learning, and thus would be a good deal more challenging, especially if a human-level analysis of a painting or photo were desired. With respect to our current approach, we could also have conducted a more thorough linguistic analysis and worked to define more precisely the meaning of certain words given the context of the text as a whole. For example, our program often interpreted love letters as having an overall joyful connotation rather than one characterized by love due to the similarity of the words that represent both emotions. Moreover, experimenting with alternative or more rigorous drawing techniques - like drawing specific portions of text in the same area on the canvas, especially in the event that texts can take on multiple sentiments through time - could have resulted in more varied artistic expression, but we are very pleased with the results of our current algorithm.