

# Mapping Emotional Landscapes of Fiction

## Using Machine Learning Techniques

Nick Levitt

April 21, 2016

### **1 Introduction and Motivation**

The art of storytelling has been around for as long as human communication, and has served as a medium for the transfer of cultural, religious and entertainment-based ideas. One of the most important aspects of storytelling is the emotion conveyed to the audience. With spoken language, the listener's perception of emotion is largely based on human context clues: facial expressions, body language, volume, etcetera. In written language however, none of these clues exist; there are just words on a page. This experiment is concerned with how and why these emotional perceptions still occur sans human context.

This experiment set out to try to build a model that can learn and

predict emotional sentiment based solely off text. This task will become increasingly important as the field of artificial intelligence progresses, as recognizing emotion is one of the most important aspects of human-human communication. If a model can be trained to predict emotion experienced based off only written words, then theoretically that model could be applied to works of literary fiction to map the emotional landscape of a story. I employed multiple machine learning techniques to simulate the perception of emotion in written language, more specifically, the perception of joy, fear, anger, sadness, disgust, shame and guilt in works of literary fiction, and visualize the results.

## **2 Background**

### **2.1 Related Works**

Sentiment analysis of text is not a new concept, and a plethora of previous research has been done in the field. Unfortunately, most of this research has been focused on distinguishing between positive and negative sentiment, the analysis of more specific emotions is less common. [Das and Bandyopadhyay 2011]<sup>1</sup> analyze broad emotional sentiment of statements in

their research using a support vector machine, but do so using multiple predictor variables, such as intensity and longevity of the emotion. [Y. Kim 2014]<sup>2</sup> implements a convoluted neural net on top of Google's Word2Vec module to classify sentiment in sentences, but once again is primarily focused on distinguishing between positive and negative sentiment.

This experiment builds upon the work of both [Das and Bandyopadhyay 2011] and [Y. Kim 2014]. All models are trained and tested on the same dataset used by [Das and Bandyopadhyay 2011], but uses only two of the features, rather than a collection of features. Also, the idea of employing a support vector machine was pulled from this paper. Similarly, the idea of applying a convoluted neural net came from [Y. Kim 2014].

## **2.2 Skills**

Existing skills I had prior to the start of this experiment included a basic understanding of Python syntax, knowledge of machine learning techniques including bag-of-words, support vector machines, cross-validation and data manipulation. I had previous experience with coding in R and building interactive applications with R-Shiny, so once I had results the visualization process went smoothly. However, I had next to no previous experience with natural language processing, and had worked primarily with numeric data

before this experiment. As a result, I had to develop a new skill-set for manipulating textual data, which turned out to be more of a challenge than I expected. The biggest challenge for me was learning how to implement all the different Python modules that I employed for this experiment. These include:

- *TensorFlow*<sup>3</sup> – used to build the convoluted neural net
- *scikit-learn*<sup>4</sup> - used to build the support vector machine
- *Pandas* – used to structure and manipulate the data
- *Numpy* - used to manipulate the data

## **3 Methods**

### **3.1 Data**

The International Survey On Emotion Antecedents And Reactions (ISEAR)<sup>5</sup> is a study done by Affective Sciences in which over 3000 persons were asked to report on situations in which they had experienced one of seven major emotions, joy, fear, anger, sadness, disgust, shame and guilt. The dataset is composed of 7666 individual observations. Each observation consists of the emotion experienced, a brief recount of the situation, and

over 40 other variables such as sex, age and country of the reporter, intensity and longevity of the emotion felt and reaction to the situation.

This experiment used only on the emotion experienced and the testimony of the situation. All testimonies are told in first-person and rarely span more than three sentences:

- “At the butcher's I saw an animal which had just been slaughtered; blood was dripping on the floor.”
- “New year's eve 1983/1984, I met my girlfriend. We stood on the steps outside her parents home and I kissed her for the first time.”

All models for this experiment were coded for in Python and trained and tested using the ISEAR dataset.

### **3.2 Naïve Bag-Of-Words**

The first model tested was a bare-bones bag-of-words, trained on two-thirds of the data and tested on the remaining one-third. The algorithm partitions the ISEAR dataset by emotion and parses through each individual emotional dataset creating a dictionary of counts for every unique word. The algorithm then parses through each individual sentence in the testing set and, using the dictionary of counts, tallies how many times each word appeared in each

emotional dataset. The sentence as a whole is then predicted to be whichever emotion occurs most frequently. Pre-processing for this model includes splitting the sentences so they can be inputted as arrays of individual words (tokenization), as well as the removal of stop-words such as “I”, “me”, “in”, “and”, “to”, etcetera.

### **3.3 Support Vector Machine**

The second model tested was a support vector machine (SVM). Using the scikit-learn module a pipeline is built to tokenize the situations from the ISEAR dataset and build a dictionary of counts, transform the counts from occurrences to frequencies if applicable, and finally build a classifier using the training data. The SVM is trained and tested using a 10-fold cross validation, with parameter tuning for determining if the model should be trained on every word individually or on pairings of two adjacent words, if the model should transform counts to frequencies, and if the model should include a penalty parameter of either .001 or .01. Preprocessing for the SVM includes the removal of stop-words.

### **3.4 Convolved Neural Net**

The third model tested was a convoluted neural net (CNN). The code is built upon an initial model designed by Denny Britz and made available on

Github<sup>6</sup>. Most major changes made to the code for this experiment centered around updating the model to run on multi-class instead of two-class data, as well as updating the initial dictionary built and changing the training/testing criteria. The CNN is an implementation of the TensorFlow module and is based loosely off of [Y. Kim 2014]. The model was trained on two-thirds of the data and tested on the remaining one-third.

## 4 Results

As can be seen from Figure 1, the SVM outperformed the other two models with an accuracy rate of 0.580, roughly four times better than random chance ( $\sim .143$ ). As a result, all further testing and visualization has been done using only the SVM model.

ACCURACY		
BAG-OF-WORDS	SVM	CNN
0.470	0.580	0.511

Table 1: Accuracy of Three Machine Learning Techniques

Some of the results are interesting from a semantics point-of-view. For example, as can be seen in Figure 2, joyful statements are misclassified more often as sadness than as any other emotion, and vice-versa. Shame and guilt are the most often misclassified emotions, both most commonly misclassified as anger. This may show an inter-relatedness between these three emotions in terms of how they are viewed from a first-person point of view; shame and guilt may be a sort of anger at the self.

JOY MISCLASSIFICATION RATE (0.261)					
FEAR	ANGER	SADNESS	DISGUST	SHAME	GUILT
0.203	0.217	0.377	0.101	0.043	0.058
FEAR MISCLASSIFICATION RATE (0.327)					
JOY	ANGER	SADNESS	DISGUST	SHAME	GUILT
0.281	0.229	0.198	0.146	0.063	0.083
ANGER MISCLASSIFICATION RATE (0.463)					
JOY	FEAR	SADNESS	DISGUST	SHAME	GUILT
0.183	0.206	0.238	0.183	0.103	0.087
SADNESS MISCLASSIFICATION RATE (0.477)					
JOY	FEAR	ANGER	DISGUST	SHAME	GUILT
0.280	0.243	0.229	0.075	0.056	0.047
DISGUST MISCLASSIFICATION RATE (0.463)					
JOY	FEAR	ANGER	SADNESS	SHAME	GUILT
0.181	0.153	0.386	0.130	0.107	0.042
SHAME MISCLASSIFICATION RATE (0.648)					
JOY	FEAR	ANGER	SADNESS	DISGUST	GUILT
0.190	0.151	0.310	0.120	0.120	0.108
GUILT MISCLASSIFICATION RATE (0.657)					
JOY	FEAR	ANGER	SADNESS	SHAME	DISGUST
0.142	0.133	0.357	0.174	0.084	0.110

Table 2: Misclassification Rates by Emotion

## 5 Visualization



The final step of this experiment was to take the most successful machine learning technique and apply it to different work of literary fiction. All novels and stories were taken from Project Gutenberg<sup>7</sup>, a digital library that offers free ebooks for download.

In order to visualize the emotional landscapes, the SVM was first retrained on the entire ISEAR dataset in order to be as robust as possible. Fictional works were then partitioned into fifty equi-length sections and passed to the SVM. The output is a numerical representation of how strong the model perceives each of the seven emotions to be in each of the fifty sections, which can then be graphed. All graphing was done using the ggplot2 package for R. In order to make the visualizations more interactive, an R-Shiny application was built so a user can pick any of the available stories and choose any subset of the seven emotions to be visualized (see Figure 1). As can be seen in Figure 2, on their own as data points these values don't appear to give any information, so a line graph joining them is built. The basic line graph of these values tends to be overly-acute and hard to follow though, so a loess function is applied to smooth out the plots.

Figure 1: R-Shiny Interactive Visualization Application

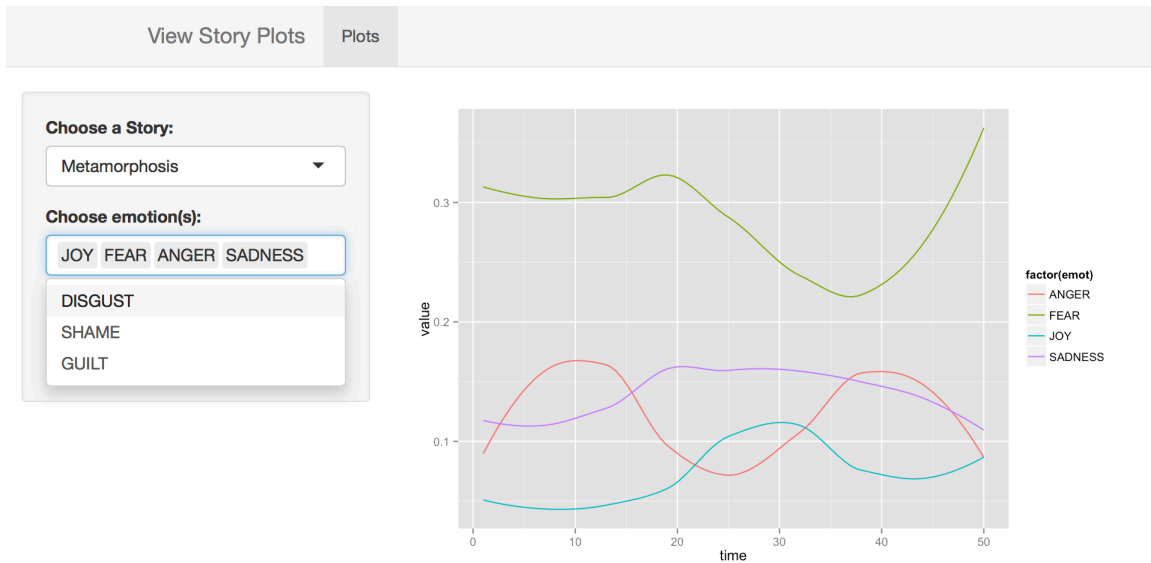


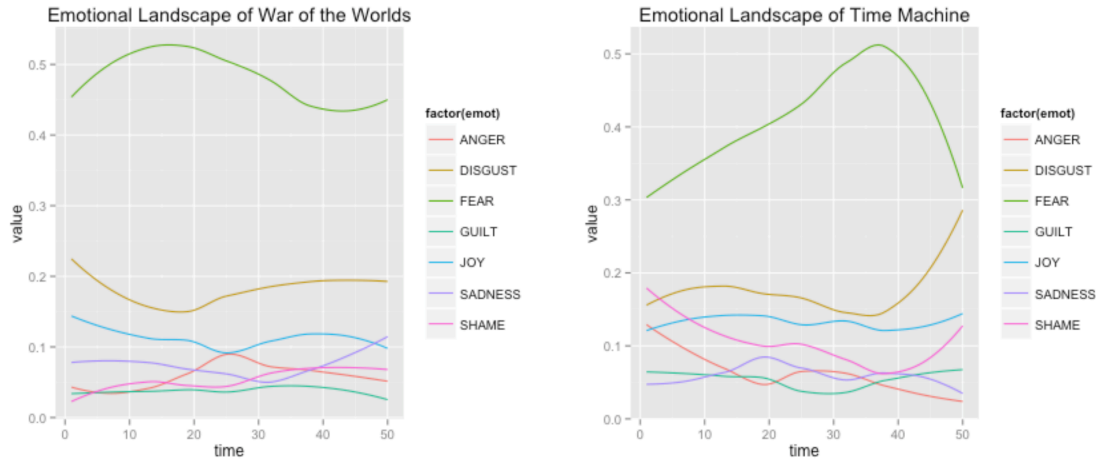
Figure 2: Visualization Process



With the visualizations, the writing styles of different authors can be viewed in a unique way. For example, it can be seen in Figure 3 that H.G. Wells instilled in both *War of The Worlds* and *Time Machine* an

overwhelming amount of fear and a secondary serving of disgust, with the 5 other emotions maintaining rather low intensities through the stories.

Figure 3: Visualization of H.G. Wells Novels



As would be expected, stories that are widely considered to be more complex, such as *Anna Karenina* by Leo Tolstoy or *Alice in Wonderland* by Lewis Carroll, have more complex visualizations, with emotions overtaking one another for prominence in any given section of the story.

## 6 Discussion and Future Work

This experiment gives an interesting insight into the abilities of different machine learning techniques when tasked with multi-class sentiment

analysis. The convoluted neural net as a technique has been garnering a lot of attention and support recently for its ability to complete a myriad of tasks, but this experiment shows that there really is no free lunch. This visualization of emotional landscapes can allow for novel analysis of both stories themselves and the writing styles of the authors who pen them. These plots are an interesting tangible representation of emotional perception, a usually personal experience.

There are many ways this experiment could be improved upon, some more intensive than others. Obviously it would be ideal to raise the accuracy of the models when testing on the ISEAR dataset. This could be done by employing more pre-processing of the text before passing it to the models. For example, part-of-speech tagging could add more insight, as could looking at the text as trigrams rather than as individual words or bigrams. It would also be interesting to find a different dataset with labeled emotional text, preferably one that is not composed of only first-person testaments. As not all fiction is written in first person, it would be more appropriate to train a model on a more diverse dataset. Lastly, it is hard to validate the results of the visualizations because perception of emotion in literary fiction is an amorphous concept, it changes from reader to reader. Whereas one reader may view *Alice in Wonderland* as a lighthearted children's story, another

may view it as a darker, more adult-centric novel. There is not much that can be done about this outside of going through multiple novels section by section and labeling their emotions in a supervised manner to be used in testing the machine learning algorithms.

## **7 Bibliography**

1. Kim Yoon. 2014. Convolutional Neural Networks for Sentence Classification. *EMNLP* (2014), 6 pages. DOI: <http://arxiv.org/abs/1408.5882>
2. Dipankar Das, Sivaji Bandyopadhyay. 2011. Analyzing Emotional Statements – Roles of General and Physiological Variables. In *Proceedings of the 5<sup>th</sup>. IJCNLP 2011 Workshop on Sentiment Analysis where AI meets Psychology (SAAIP)*. Chiang Mai, Thailand, 59-67. DOI: <http://www.aclweb.org/anthology/W11-37>
3. Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, Sherry Moore, Derek

Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

4. Cournapeau, David, and Et Al. *Scikit-learn*. Computer software. *Scikit-learn*. Vers. 0.17.1. N.p., n.d. Web. Spring 2016.
5. *International Survey On Emotion Antecedents And Reactions (ISEAR)*. Affective Sciences, n.d. Web. Spring 2016.
6. Britz, Danny. *Cnn-text-classification-tf*. Computer software. N.p., n.d. Web. Spring 2016.
7. "Project Gutenberg." *Project Gutenberg*. N.p., n.d. Web. Spring 2016.

