# Movie Review Visualization

Dan Prendergast*

University of Colorado Boulder

## ABSTRACT

Interpreting subtle meaning and sentiment in conversation is one of the most complex cognitive tasks in which humans engage. Having a real time sentiment meter could help people assess how a conversation is going. Techniques in natural language processing (NLP) have been applied to sentiment analysis in text and speech domains, but top performing systems use deep neural nets or expensive statistical methods. These would require significant processing power and may not execute in real time. In this project, we visualize multiple, simple-to-encode metrics of the movie reviews to see if any features of the reviews permit fast discrimination of review sentiment. In this project, we visualize several features to see if they could be used for sentiment analysis in real-time. Unfortunately, none of the visualizations indicate potential for sentiment analysis using simple-to-measure features of movie reviews.

**Keywords**: Movie review, sentiment analysis, word embedding.

**Index Terms**: N/A

## 1 INTRODUCTION

Interpreting subtle meaning and sentiment in conversation is one of the most complex cognitive tasks in which humans engage. The ambiguities of terms, analogies and metaphors, and sarcasm only add to the difficulty. The ability to interpret sentiment drives the course of a conversation. It can mean the difference between a pleasant conversation and verbal conflict.

Having a real time sentiment meter could help people assess how a conversation is going. The speaker could adjust their tone or topic to save a degenerating conversation. Techniques in natural language processing (NLP) have been applied to sentiment analysis in many domains, but top performing systems use deep neural nets or expensive statistical methods. These require significant processing power and may not execute in real time.

In this project, we visualize multiple, simple-to-encode metrics of the movie reviews to see if any features of the reviews permit real-time discrimination of review sentiment. Of course, any results found on a movie review dataset do not necessarily translate to general conversation; however, this project may provide an initial proof of concept on readily available movie review datasets. Further work would involve applying the same processing and visualization to a dataset of general conversation with sentiment annotations.

## 2 RELATED WORK

Sentiment analysis techniques are used in various domains, including product reviews, social media posts, political speeches, and media reports [1]. Techniques for solving the sentiment analysis problem range from rule-based techniques using hand labeled datasets to recurrent neural nets using automated encoding of the source documents.

### 2.1 Text Sentiment Analysis

A significant body of research exists covering sentiment analysis on text datasets. The techniques can be broadly divided into machine learning and lexicon-based approaches. Lexicon-based approaches rely on initial hand-labeled datasets associated words with emotions or valences and parts of speech. Researchers used this lexicon-based approach in [2], [3], and [4]. Qiu and He [2] used a sentiment dictionary combined with a rules-based approach to determine reviewer sentiment. Xu and Liao [3] used a sequence learning technique, called Conditional Random Fields, to learn sentiment labels for sentence substrings.

Machine learning based approaches to sentiment analysis include [5], [6], and [7]. Socher et al. [7] present one of the most successful sentiment analysis systems. It utilizes word embeddings as features and a recurrent neural network to classify sentiment of each sentence fragment within a document.

### 2.2 Real-time and Speech Sentiment Analysis

The body of work in real-time sentiment analysis is much less robust, particularly when applied to speech. Goel, Gautam, and Kumar [8] demonstrate real-time sentiment analysis of Twitter messages using a Naïve Bayes classifier; however, their system depends on the SentiWordNet database of words tagged with sentiment. Bertero et al. [9] performed real-time dialog sentiment analysis using a trained Convolutional Neural Network. Again, this solution required a speech dataset with each sentence tagged with its sentiment.

* daniel.prendergast@colorado.edu

## 3 Discussion

In this project, we hope to overcome the difficulties of approaches described above by utilizing simple-to-encode features, or features obtained without annotated datasets. I present visualizations of these features to provide an initial analysis of their feasibility for sentiment prediction. The features used in the following visualizations are mean number of words per review and word embedding representations of the reviews.

Word embeddings are trained using unsupervised learning and capture semantic information about words based on their proximity to other words over a large corpus [10]. The word embedding is a vector representation of a word. The embedding space is high-dimensional – 50 dimensions in the case of the dataset used for this project – but the space can be visualized by employing t-distributed Stochastic Neighbor Embedding (t-SNE) [11] to reduce the data to two dimensions.

Two datasets were downloaded create the visualizations. The first dataset was the "Sentiment Analysis on Movie Reviews" dataset available from the Kaggle website [12]. It is a collection of movie reviews from the Rotten Tomatoes rating site. The dataset contains the review text and a sentiment labeled with an integer from 0-4, with "0" being a "negative" review. The second dataset was a set of pre-trained set of word embeddings obtained using the GloVe algorithm [13].

To create the visualizations, the data requires pre-processing. Reviews are tokenized, stopwords (i.e., non-essential words) removed, and all text converted to lowercase. The second pre-processing step is to run the t-SNE algorithm on the set of pretrained word embeddings. This replaces the 50-dimensional embedding for each word with a 2-dimensional embedding, allowing us to plot each word in a visualization.

### 3.1 Mean Number of Words per Review

The first feature of the reviews that we examine is the mean number of words per review. My hypothesis is that reviewers giving negative marks for a movie will provide a justification for the rating, resulting in longer reviews for negative sentiments.

This metric was calculated by filtering the review dataset to get 5 subsets of the data – one for each sentiment value. Then we total number of words and divide by the number of reviews for each subset. Results are displayed in Figure 1 below.

As we can see, there is no significant correlation between the review sentiment and the number of words in the review. It is unlikely we can use this as an easy feature for sentiment analysis.

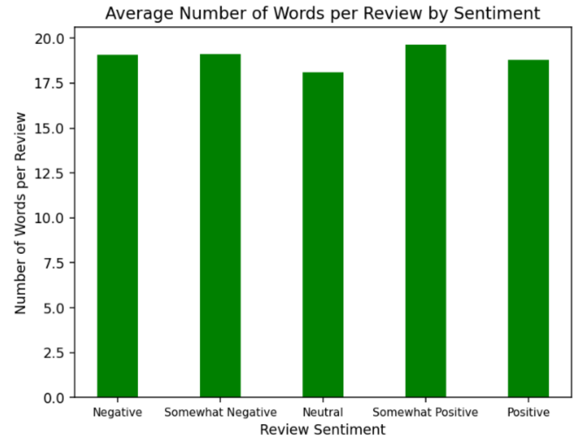


Figure 1: Mean words per review by sentiment.

### 3.2 Review Composite Embedding

Next, I will examine if a composite embedding permits discrimination of sentiments in the movie reviews. Multiple methods exist for combining multiple word embeddings into a composite embedding (e.g., addition, element-wise multiplication, averaging, etc.). In this project, we calculated a composite embedding value, in t-SNE space, by averaging the t-SNE embeddings for all the words used in each review. For this visualization, we limit the sentiments to only "positive" and "negative" reviews. The reviews labeled "somewhat positive" and "somewhat negative" were not included to try to increase separability of the two sentiment datasets. The result is shown in Figure 2 below.

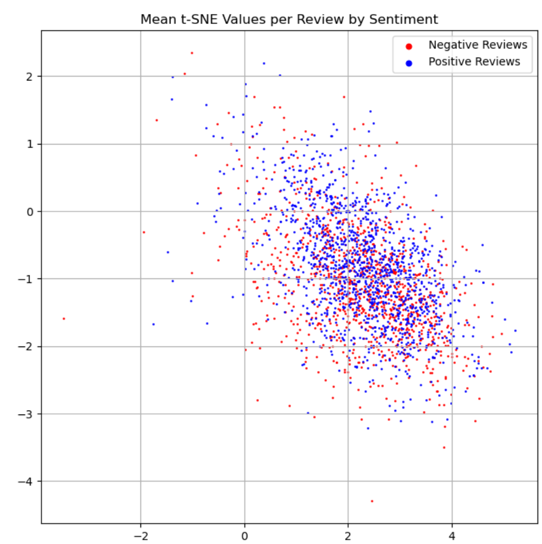


Figure 2: Mean t-SNE embedding for each review.

Unfortunately, the distributions of the positive and negative review embeddings are very similar. Using this feature as an easy discriminator of sentiment shows little promise.

### 3.3 Review Trace through Embedding Space

The last feature I would like to investigate is a trace of the reviews through t-SNE embedding space. For this visualization we will again limit the dataset to only the "positive" and "negative" reviews. To create the visualization, t-SNE embeddings are plotted for each word in the reviews and lines plotted between adjacent words in the review. The result is shown in Figure 3 below, and the interactive chart can be viewed using the Jupyter notebook included with the project submission.

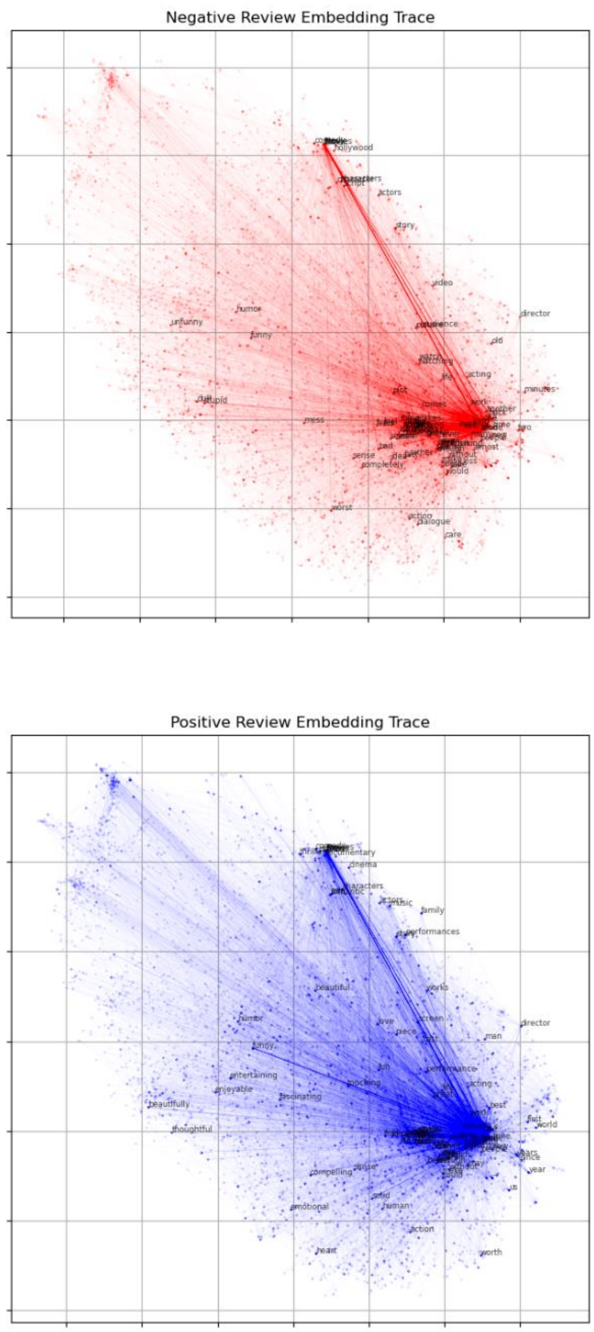


Figure 3: Two boxes. One filled with confetti.

In this visualization we see that the distributions of the words used in both positive and negative reviews is very similar. If we look at the lines indicating adjacent words in the reviews, we do see some differences in the word pairs used. This seems intuitively correct, since many of the high performing sentiment analysis systems use word sequences, instead of individual words.

## 4 Conclusion

In this project, we examined several easy-to-encode features of movie reviews to examine their feasibility as sentiment discriminators. Those features were mean number of words per review, mean t-SNE embedding of each review, and a trace of the words in each review through t-SNE embedding space. Unfortunately, the visualizations of these features do not indicate that they would be good individual features for real-time sentiment analysis. The review trace through embedding space did show some differences between positive and negative reviews by examining word pairs, as shown by lines in the plot. Future work would involve exploring these word pairs as potential features for real-time sentiment analysis.

## References


[1] W. Medhat, A. Hassan, and H. Korashy. "Sentiment analysis algorithms and applications: A survey." Ain Shams engineering journal 5.4 (2014): 1093-1113.

[2] G. Qiu, et al. "DASA: dissatisfaction-oriented advertising based on sentiment analysis." Expert Systems with Applications 37.9 (2010): 6182-6191.

[3] K. Xu, et al. "Mining comparative opinions from customer reviews for competitive intelligence." Decision support systems 50.4 (2011): 743-754.

[4] F. Cruz, et al. "'Long autonomy or long delay?' The importance of domain in opinion mining." Expert Systems with Applications 40.8 (2013): 3174-3184.

[5] C. Chen, and Y. Tseng. "Quality evaluation of product reviews using an information quality framework." Decision Support Systems 50.4 (2011): 755-768.

[6] A. Duric and F. Song. "Feature selection for sentiment analysis based on content and syntax models." Decision support systems 53.4 (2012): 704-711.

[7] R. Socher, et al. "Recursive deep models for semantic compositionality over a sentiment treebank." Proceedings of the 2013 conference on empirical methods in natural language processing. 2013.

[8] A. Goel, J. Gautam, S. Kumar. "Real time sentiment analysis of tweets using Naive Bayes." 2016 2nd International Conference on Next Generation Computing Technologies (NGCT). IEEE, 2016.

[9] D. Bertero, et al. "Real-time speech emotion and sentiment recognition for interactive dialogue systems." Proceedings of the 2016 conference on empirical methods in natural language processing. 2016.

[10] Y. Bengio, et al. "A neural probabilistic language model." The journal of machine learning research 3 (2003): 1137-1155.

[11] G. Hinton and S. Roweis. "Stochastic neighbor embedding." NIPS. Vol. 15. 2002.

[12] Kaggle. "Sentiment Analysis on Movie Reviews". https://www.kaggle.com/c/sentiment-analysis-on-movie-reviews

[13] J. Pennington, R. Socher, and C. Manning. "Glove: Global vectors for word representation." Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.