The Importance of Introduction Structure in Determining the Helpfulness of Amazon Software Reviews

February 23, 2015

Douglas Chan

1 Introduction

It is not uncommon for people studying marketing to be taught the importance of the initial and final point made when giving a presentation. The reasoning behind this is that people are more likely to remember the first part of the presentation and the very end of the presentation. Psychology classes teach the same idea. When given a list of 40 words to memorize, people were noticeably more likely to remember the words at the beginning of the list and the words at the end of the list than any of the words in the middle, according to an article by Bennet Murdoch in the November 1962 issue of the Journal of Experimental Psychology. The goal of this experiment was to determine if that same principle would also apply to software reviews taken from the website Amazon.com.

This report explores the extent to which the text at the beginning and end of a review compares to the text in the middle. Features such as word choice and sentence structure were examined to find which ones were most relevant to predicting a review's perceived helpfulness.

The following report will be structured as follows: section 2 will discuss the dataset used for analysis in this paper. Section 3 will discuss similar papers from which this paper drew ideas. Section 4 will explain the predictive task that this project attempts. Section 5 will discuss the preliminary analysis that led to the design of this experiment. Section 6 will explain the methodology used to run the experiment, as well as the thought process that went into building the final classifier. Section 7 will describe the results and conclusions that will be drawn from the data.

2 Dataset

The dataset used for this analysis was the dataset of Amazon software reviews. This dataset was chosen because it contained a sizeable number of data points (95,084) while not being too prohibitively large that running experiments on it on an average computer would not be excessively slow. It was also chosen because each datapoint has a distinct text representation that can be analyzed based on the words within it.

3 Relevant Literature

A paper by Cristian Danescu-Niculescu-Mizil, George Kossets, John Kleinberg, and Lillian Lee entitled *How Opinions are Received by Online Communities: A Case Study on Amazon.com Helpfulness Votes* lists several factors that may affect the perceived helpfulness of an Amazon review. Although the paper acknowledges that the actual deciding factor in determining the helpfulness of reviews may be the actual text content of the review, the paper proposes many non-textual factors that may play into the perceived helpfulness of a review. They hypothesize that a review that gives a star score closer to the concensus rating of the product will be considered more helpful. They also hypothesize that a person is more likely to find a review helpful if it agrees with their personal view of the product. Finally, they cite a paper by Amabile that states that negative reviewers are perceived to be smarter than positive reviewers.

In their study, they analyze a set of roughly 4,000,000 book reviews from Amazon, coming from roughly 165,000 book. More than a million of the reviews they chose contained at least ten helpfulness votes. They defined the *helpfulness ratio* of a review to be the fraction $\frac{a}{b}$ where a is the number of people who voted that the review was helpful and b was the number of people who voted on the helpfulness of the review. The same terminology will be used in this report.

Based on their experiments, they concluded that the how much a review deviates from the mean of the scores given by all reviewers is inversely correlated with the perceived helpfulness of a review. That is, the helpfulness ratio of reviews decreases as the difference between a review's score and the mean score for that product increases. In addition, when they compared reviews with a similar textual structure, they found that reviews with larger absolute deviations with similar "plagiarized" reviews with smaller deviations, they found a statistically significant difference in the helpfulness ratios of the reviews. By comparing very similar reviews, they were able to control for the content of the text and compare the helpfulness ratio based on only the deviation of the score from the mean and not from the content of the review itself. From this information, they concluded that the deviations from the mean had an effect on the perceived helpfulness of the review, and that this effect did not just happen coincidentally as a result of features in the text.

4 Predictive Task

The goal of this experiment is to explore the effect that the structure of the text of a review has on the perceived helpfulness of that review. The inspiration for this predictive task came from a presentation given during a faculty research seminar given during the fall quarter of 2014 at the University of California, San Diego. The speaker presented his research on the language of Yelp restaurant reviews, and concluded that people who enjoyed eating at expensive establishments were prone to using sexual imagery in their reviews (e.g. the food was orgasmic) whereas people who enjoyed eating at cheap establishments were more prone to using drug imagery (e.g. the person is addicted to the food). The professor also investigated posts to the random acts of pizza subreddit, a forum on the website reddit.com in which people occasionally sent other people pizza. He tried to analyze what language and structure of posts was most likely to result in a person giving the poster a pizza. For example, if a poster claimed that he was going to pay the act forward at a later date, he was more likely to receive a pizza.

This predictive task attempts to do a similar analysis to Amazon reviews, instead of measuring whether a post will receive a pizza, it measures what percentage of people will find the review helpful, based on the text of the review. For simplicity, this predictive task was framed as a classification problem, with the three classes corresponding to reviews with a helpfulness ratio of less than $\frac{1}{3}$, reviews with a helpfulness ratio of $\frac{1}{3}$ to $\frac{2}{3}$, and reviews with a helpfulness ratio of greater than $\frac{2}{3}$. The rationale for the decision to frame the problem this way will be given in the model section of this report. This experiment will be run on series of words from the beginning, middle, and end of the reviews, and the results will be compared to determine if certain portions of the review are more influential in determining how much a review is perceived to be helpful.

Additional inspiration for this predictive task came from the previously mentioned paper by Danescu-Niculescu-Mizil, Kossets, Kleinberg, and Lee. In their paper they acknowledge that a significant part of the perceived helpfulness of a review comes from its textual content. While textual content in itself can take many different forms, it is possible for certain word choices and paragraph structures to be more helpful than others. While these features are not as independent of the text as a whole as the features mentioned in the paper, they may serve as useful features to examine when determining what makes a review helpful. Thus, part of the goal of this predictive task was to further reveal what factors contribute to the helpfulness of a review.

The inspiration to test on substrings from the beginning, middle, and end of the string came from a study done in 1962 by Bennet Murdock. Subjects were shown 10-40 words and then asked to recall them. The results of the study showed that the first four words of the list and the last eight words were recalled significantly more than any other words in the list (Murdock, 1962). The tendencies of people to recall the first and last parts of information they are presented are known as the primacy effect and the recency effect, respectively. The application of this finding to Amazon reviews is that if viewers are more able to remember the beginning and the end of the reviews that they read, then those portions of the review may be more influential in determining whether a person will find a review helpful. The hypothesis is that since the users are more likely to remember the contents of the beginning and the end of the review, if the beginning or end of the review is structured and worded in a way that people find helpful, then that will be more indicative of whether a review will be considered helpful than the wording and structure in the middle of the review.

In order to conclude that the text of a review has an effect on the perceived helpfulness of the review, this experiment must show that the results of the classifier perform significantly better than the results of choosing a class at random. Random choice, assuming that the pseudo-random number generator used is random enough that it does not significantly affect the outcome of the experiment, should produces an accuracy rate of .33. That will serve as the baseline with which to compare the experimental results.

The expectation of this experiment is not that it will produce classifiers with a high accuracy. Taking a review and only looking at the beginning, middle, or end will definitely ignore some features that are instrumental in determining the helpfulness of the review. In addition, on reviews with only a few votes, a difference of one vote can be the difference between two classes. It is conceivable that two reviews with almost the exact same text could be classified as two different classes just by virtue of the fact that not as many people have voted on one as have voted on the other. This fact would lower the absolute accuracy of the classifier. As a result, absolute accuracy is not expected, besides beating the random choice baseline. Instead, the expectation is that one of the classifiers may perform significantly better than the others, thus

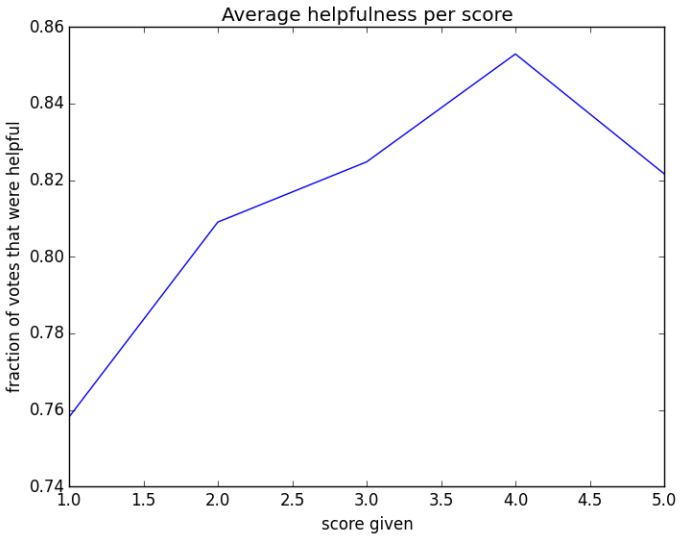
indicating that certain regions of the review are more important than the other regions.

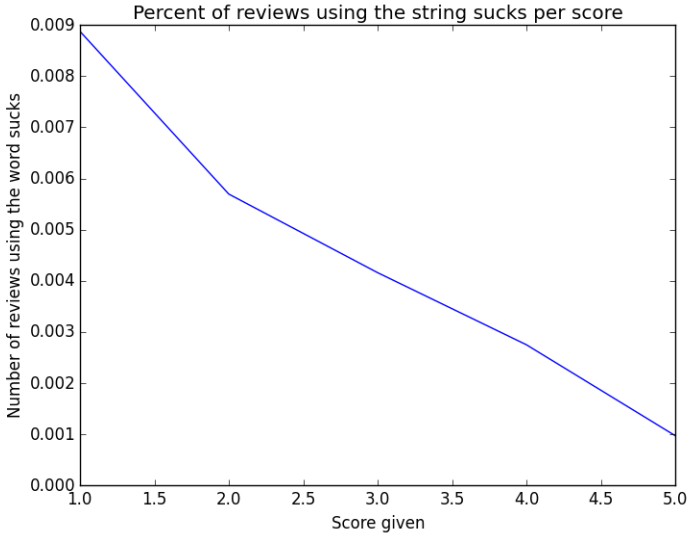
5 Preliminary Analysis

The first task was to prove that the text of a review had an effect on its perceived helpfulness. One aspect of the text which was predicted to influence the perceived helpfulness of the review was the word choice, the words in the review itself. In order to test this, a multinomial naive bayes classifier was constructed to attempt to predict if knowing the distribution of words in a review could help classify it as helpful or unhelpful. Multinomial naive bayes is not a very accurate model, as it just looks at the distribution of words and not the order, and assumes that all words are independent of each other. However, this also means that if the multinomial naive bayes classifier performs better than just randomly choosing a class, it means that the word distribution has some effect on the perceived helpfulness of a review.

As simple multinomial naive bayes built to classify reviews as either helpful (having a helpfulness ratio of .5 or greater) or unhelpful (having a helpfulness ratio of less than .5) correctly predicted the class of reviews with a 65% accuracy. This significantly outperformed the predicted 50% accuracy from guessing classes randomly. This result in itself suggested that textual wording of a review may have a significant impact on its perceived helpfulness.

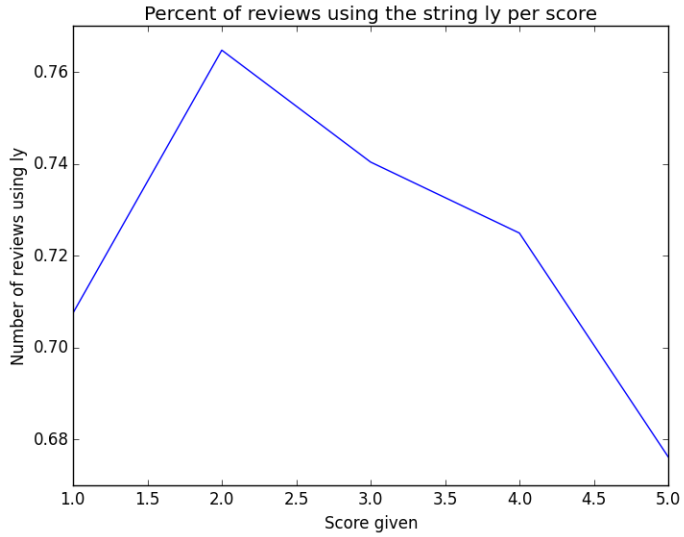
Further analysis was done to prove the correlation between the language and the helpfulness of a review. This was mostly done by plotting the average helpfulness ratio per score and the percentage of reviews that used certain words per score.





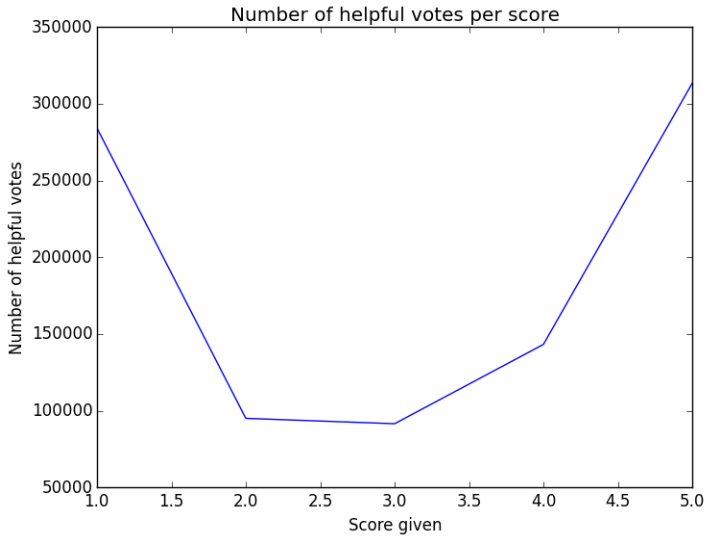
In one simple example, the percentage of reviews that used the word "sucks" was plotted for every score. As expected, the percentage spikes at one star reviews. From the first graph, it can be observed that one star reviews on average were found the least helpful. These results are not trying to say that the presence of the word "sucks" makes a review seem less helpful, but it is plausible that they may have a similar cause and therefore maybe be able to help predict each other.

In another experiment, the percentage of reviews that contained the string "ly" was plotted for every score. "Ly" is a component of a large subset of adverbs, so the absence of the string may coincide with very limited use of adverbs, which would speak to the structure and style of the review. While the presence of "ly" may just mean that the word "lying" was used, for example, the absence requires the absence of many adverbs.



As the graph shows, the percentage of reviews with the string "ly" varies in a significant way depending on score. This suggests that some scores may be more likely to utilize adverbs, and that since helpfulness and score are related, the wording and structure of a review may affect its helpfulness.

Finally, the relationship between number of helpfulness votes (the number of people who voted either that a review was helpful or unhelpful) was plotted for every score. Results of this graph are inconclusive, as there are likely more one and five-star reviews than any other scores.



6 Model

All points were given one of three classes. The first class corresponded to the reviews that received a helpfulness ratio of between .67 and 1.0. The second class corresponded to the reviews that received a helpfulness ratio between .33 and .66. The third class corresponded to the reviews that received a helpfulness ratio of less than .33. This three class system was created to create discrete classes so that 1-nearest neighbor could be used to classify points.

The final form of the classifier used 1-nearest neighbor to classify the elements in the test set by comparing them to elements in the training set. The metric used to compare the distance between two points was the edit distance of the text strings of the reviews. This metric is also known as the Levinshtein distance. The Levinshtein distance of two sequences is the minimum number of insertions, deletions, or edits that need to be made so that the two sequences become identical. The edit distance was chosen as a metric because it encodes aspects of word order as well as aspects of which words are used in the review. Two sequences of the same words in different orders will have a higher edit distance than two sequences of the same words in the same order. Two sequences of words with the same sentence structure and mostly the same words but with a few words changed will have a higher edit distance than they would if the words were not changed.

The edit distance is a relative quantity between two points rather than an absolute quantity. Therefore, since the goal of the experiment was to unearth similarities between reviews that may lead to them being considered helpful, nearest neighbor became the natural choice for the classifier. If text structure is an influential feature in determining helpfulness, then in theory similarly structured review should have a similar helpfulness. Reviews with a smaller edit distance will be more similar in their structure because of how edit distance works. As a result, reviews that have a small edit distance would likely have a similar helpfulness, and nearest neighbor encodes that idea perfectly.

Using nearest neighbor and edit distance had its drawbacks, however. Edit distance is extremely slow. When implemented using dynamic programming in the way that is taught in CSE 101, the running time is $O(nm)$, where n and m are the lengths of the two strings. Amazon reviews, however, could reach thousands of characters long, making calculating a sufficient number of edit distances to run nearest neighbor prohibitively slow. To compromise, instead of calculating the edit distance of the review strings, the edit distance of the sequence of words within the reviews were calculated instead. This change reduced the length of each review from thousands of elements to hundreds of elements, significantly speeding up calculations. In order to make this happen, the string had to be split by whitespace. Python has a built-in function for this.

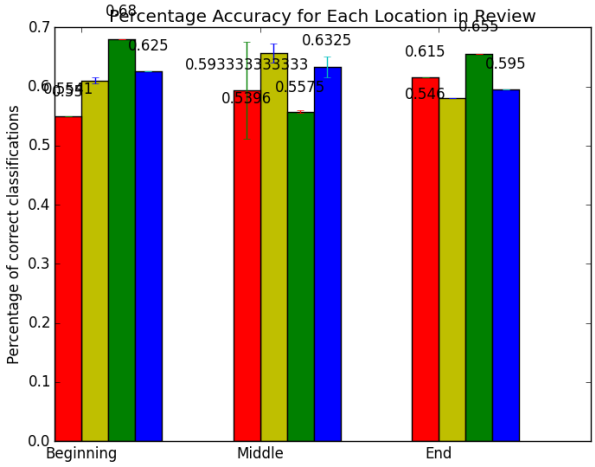
However, calculations were still extremely slow. Even with aggressive multithreading, calculating the edit distance between one test point and sixty thousand training points took around 45 seconds on a desktop that was custom-build for performance. That meant that if classifying a test set of 10,000 points was

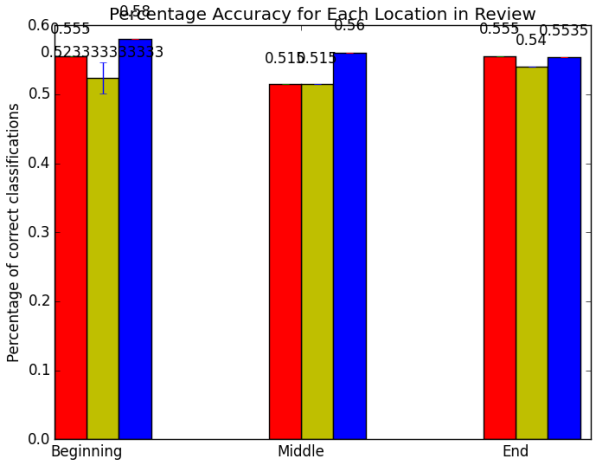
attempted, the resulting calculations would take multiple days to complete. Given the finite amount of time before the due date of this project, that seemed infeasible. Even given that the experiment only called for comparing substrings from the beginning, middle, and end of the reviews, even with reduced sized substrings the projected time stretched into days of calculation per experiment.

The idea of using nearest neighbor and edit distance was almost discarded because of how slowly it ran. Some preliminary tests that tried using multinomial naive bayes classifiers trained on different sections of the review were run. For example, a multinomial naive bayes classifier was constructed and trained on only the first 25 words of each review. Test reviews were then classified based on their first 25 words, and a class was chosen based on the word distribution. However, for the three-class model, these classifiers performed abysmally, averaging a 27% accuracy rate, worse than randomly choosing classes. The small subsets of the reviews were not enough to provide an accurate model of the word distribution in reviews. Additionally, naive bayes classifiers assume that words are independent, and thus do not encode any aspect of word order, thus potentially missing out on a very important feature. At this point, the decision was made to go with the classifier that most closely fit the hypothesis instead of the one that would be able to produce more data points. As a result, the idea was discarded, and the experiment began running the initial nearest neighbor and edit distance plan, albeit with severely restricted test and training set sizes.

The following graphs display the results of testing the classifiers. Although they are simple and may not look like much, they are the result of more than 35 hours of computer computation, just running the same program over and over with small tweaks to parameters.

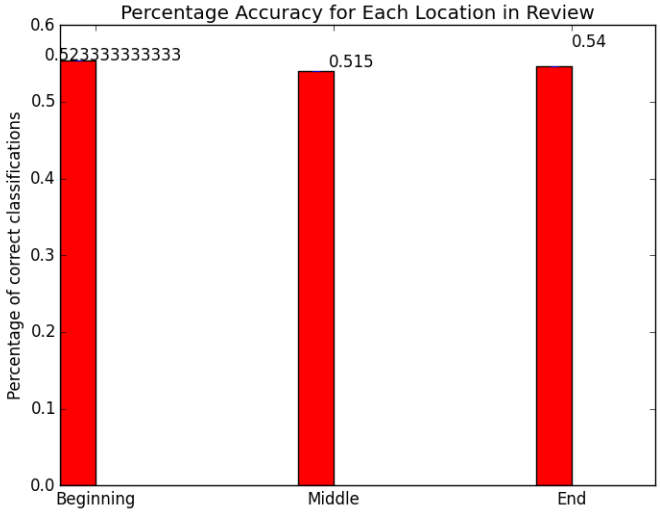
The first graph shows the mean accuracy of the classifier when trained on 500 randomly selected points and tested on 200 randomly selected points, using different sized subsets from the beginning, middle, and end of reviews. Points that did not have any helpfulness votes were thrown out. Luckily, this still left over 60,000 points with which to work. The left-most red bar represents the mean accuracy when the subset size is restricted to ten words. The yellow bar represents the mean accuracy when the subset size is restricted to 25 words. The green bar represents the mean accuracy when the subset size is restricted to 40 words, and the blue bar represents the accuracy when the subset size is restricted to 50 words. A legend would have been added, but it blocked the data.





The graph next to it represents the mean accuracy when the test is run with 50,000 randomly selected training points and 200 randomly selected testing points. Because of time constraints, results for subsets of size 40 were unable to be generated.

Finally, to test if small number of test points was messing up the data, the experiment was run for a training size of 10,000 and a test size of 500, with the subset size being restricted to 25 words.



7 Results and Conclusions

From the data, it can be concluded that text structure and wording do indeed affect the perceived helpfulness of reviews. None of my 3-class classifiers ever reported less than a 50% accuracy in classifying data points, even with small training sizes, making them significantly more accurate than randomly choosing classes.

With the current number of results collected, it is impossible to conclude whether the beginning or the end of the review is more important than the middle of the review. The trials run with a training size of 500 data points have conflicting results. For a subset size of 50, the middle of the review provided a more accurate classifier than the beginning or the end, but for other sizes the classifiers based on the text at the beginning or the end of reviews outperformed the classifiers based on the text in the middle of the review.

For the tests run with 50,000 training points and 200 testing points, the middle classifier was never the most accurate classifier. Either the beginning or the end of the review was more influential than the middle, if only slightly. These results cannot be considered statistically significant, however, until more trials are run.

Increasing the test set size to 500 and decreasing the training set size to 10,000 does not seem to have affected the accuracy rate by very much. It seems like the test set size was large enough in both cases to get a good idea of the accuracy of the classifier.

8 Future Work

A larger number of trials must be run in order to determine the statistical significance of the data. Time constraints precluded this experiment from being able to conclude anything from the trends expressed in the data.

Furthermore, trials must be done with a larger number of nearests neighbors. 1-nearest-neighbor is not robust against noise, as a mislabeled point may cause many other points to become mislabeled. Increasing the number of neighbors considered would increase the robustness of the classifier by requiring more mislabeled points before it has an effect on the data and points are misclassified as a result. Time constraints, however, resulted in the use of 1-nearest-neighbor.

Additional methods of pre-processing the data may yield useful results as well. The decision was made in this experiment not to filter out common words, because of how edit distance is affected by word order as well as word choice. However, in future trials, filtering out certain words may change the effectiveness of the classifiers.

9 Works Cited

Kossinets, George et al. How Opinions are Received by Online Communities: A Case Study on Amazon.com Helpfulness Votes. http://www.cs.cornell.edu/home/kleinber/www09-helpfulness.pdf

Pang, Bo and Lee, Lillian. *Opinion mining and sentiment analysis.* http://www.cs.cornell.edu/home/llee/omsa/omsa.pdf

Yin, Xiaoxin et al. *Truth Discovery with Multiple Conflicting Information Providers on the Web.* http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4415269

T. Amabile. *Brilliant but cruel: Perception of negative evaluators.* J. Experimental Social Psychology, 19:146156, 1983.

The serial position effect of free recall. Murdock Jr., Bennet B. Journal of Experimental Psychology, Vol 64(5), Nov 1962, 482-488. http://dx.doi.org/10.1037/h0045106

Shinzaki, Dylan et al. *Trust and Helpfulness in Amazon Reviews: Final Report* . http://snap.stanford.edu/class/cs224w-2013/projects2013/cs224w-060-final.pdf