Predicting Amazon review helpfulness

[CSE 255 Assignment 1]

Shitij Bhargava
University Of California, San Diego
shbharga@eng.ucsd.edu

ABSTRACT

Reviews on amazon are ranked by how helpful they are rated by users in an effort to quickly summarize the opinions of a product for potential buyers. This project aims to explore what factors affect a review's helpfulness by building a classification model on the Amazon movie reviews data set. The model performs well with accuracies over 85% and it is found that a review's writing style, product rating and unigram features affect helpfulness the most.

1. INTRODUCTION

Reviews written on Amazon for products have an option to be voted as either helpful or not helpful. Such options are commonly present in many online retailers so that reviews that summarize the common opinion best are shown first to potential buyers in order to make decision making easier. But currently as users have to vote on a review, there is no automated way of knowing if a review is helpful when it is newly posted. This is an issue as helpfulness measurement by voting are affected by various kinds of biases like the winner circle bias where products with higher ratings might get more helpfulness votes or the early bird bias where a review written earlier maybe be deemed more helpful by users even though a recently posted review may have much more information and analysis of the product in question. This makes identifying which reviews might be truly helpful in making a decision to buy quite hard. But this is an important problem as many people try to make an objective opinion of a product before buying by going through numerous reviews. As the number of reviews can be large they can only go through a set of reviews most visible to them. Many different studies on a variety of datasets have been performed studying various aspects of a review like linguistics features and metadata to try and model review helpfulness. This project attempts to discover if their findings hold true on this dataset and discover how much effect biases have in determining helpfulness as voted by users.

2. DATASET STATISTICS

The dataset consists of amazon movie review data taken from [10]. Each review has the fields: product ID, user ID, profile name, helpfulness, review score, review time, review summary and review text.

The following table shows some general statistics for the data:

Total number of reviews	7,911,684
Number of users	889,176
Number of products	253,059
Timespan	Aug 1997 - Oct 2012

The following table shows statistics related to helpfulness:

Mean helpfulness votes	5.521
Mean helpfulness ratio	0.606
Number of reviews with no votes	2,104,404

3. DATA PRUNING

The data needed to be prepared for analysis and it was pruned in the following ways:

1. Reviews with no helpfulness votes were removed as the helpfulness ratio does not exist for them
2. Reviews with votes less than 10 were also removed since the helpfulness ratio of reviews with very low votes is harder to compare with reviews with higher votes. This way of pruning has been followed previously [9]
3. Reviews with more than 8000 votes were removed. Except two reviews in the whole dataset, all reviews had <8000 votes, hence they were treated as outliers and removed. The box plot for resulting data can be seen in figure 1
4. For computational reasons, a random sample of 600,000 reviews was chosen out of the remaining 1,037,619

It should be noted that 'plagiarized' reviews or reviews having very similar textual content in different products were not removed.

4. DATASET EXPLORATION

The following subsections try to explore what relationship different variables in the review have with review helpfulness. many of these variables are compared with helpfulness ratio to notice any trends, but a more exhaustive

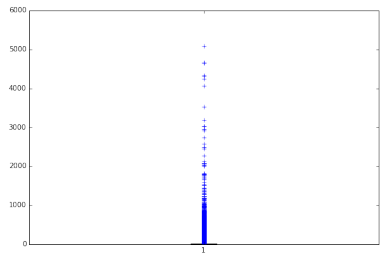

Figure 1: Box plot showing number of votes received by reviews

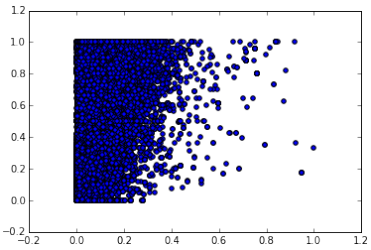

Figure 2: Scatter plot of length with helpfulness ratio

4.1 Product review score

The histogram in figure 3 suggests that most people either give a 5 star review or a 1 star review. Interestingly, this closely matches how people rate a review's helpfulness too. Most reviews are rated as either extremely helpful or not helpful at all. This can be seen in the histogram in figure 5.

Additionally, products with higher ratings tend to have higher helpfulness ratios as well, as can be seen in figure 4.

4.2 Review score deviation from mean

By conformity principle one would expect that if a review deviates more from the mean rating of a product, more people will disagree with it and find it unhelpful. The mean deviation can be absolute or signed. Absolute would measure how much a reviewers opinion differes from the rest and a signed would take into account if the reviewer deviates in the positive or negative direction. Both the correlations were calculated and are shown below:

Signed deviation correlation: 0.4668 (reviewer rating - mean rating) Absolute deviation correlation: -0.1913

Absolute deviation correlation is in line with the conformity principle, but a much higher correlation for signed deviation indicates that which direction the deviation is, is an important factor in determining if the review will be marked as helpful or not.

4.3 Review length

Character length of review text has a weak correlation with the helpfulness ratio of reviews. For the pruned dataset, this correlation came out to be 0.2741. The scatter plot is shown below:

The lengths on the x-axis are min-max scaled between 0 and 1. It can be seen that there appears to be a stronger correlation when scaled length is over 0.3.

Similarly if instead of raw character length, we use number of words the result is quite close. The correlation comes out to be 0.2695.

For length of summary a somewhat smaller correlation is present (0.1695)

4.4 Total number of votes

The total number of votes doesn't seem to have any correlation with the helpfulness ratio. The correlation was only 0.038. Thus it seems if many people are voting a review it is not necessarily because its helpful (or not helpful).

4.5 Review writer's experience

Experience here can have different kinds of meanings and a two were explored as given below. One way can be to count the number of movie reviews a user has written before writing the current review. This might reflect his experience in movies as well as review writing. Another metric can be to see the time difference of his currently written review to when he wrote the first ever review. This might reflect his experience with movies. One more metric can be to see how many total reviews a user has written.

On average a user has written 3.8318 reviews.

1. Time difference of review with first ever review by a user
   This way of measuring experience gives no correlation (-0.0344) with helpfulness ratio

2. Total number of reviews written by user
   Total number of reviews means how many reviews a user has written and will write in the future at any given point of time. This might measure how much a user is predisposed for enjoying movies and writing reviews for them. The correlation with helpfulness ratio is rather low at 0.1513

4.6 Rank/ Time delay of a review in a product

It is possible that reviews that are written first for a product will get more positive helpfulness ratios. For example, if a product is really good (or really bad) the first reviewer might be more compelled to mention that and subsequent buyers might just agree with him/ her. There are two ways we can measure this:

1. Absolute time difference of reviews in a product The time difference is measured as the difference of the current review time from the previous review written for the product. So for example, if for product P review A was written at x and review B was written at y, where x <y. The absolute time difference for A will be 0 and for B will be y-x. This measures how 'late' a review was compared to the last review for the same product.

   The correlation came out to be -0.2256 which is negative as expected, although weak. This means that

the later a review is written for a product the less it's helpfulness ratio.

2. Rank of review in a product Rank is defined as 0 for the first review to appear for the product, 1 for the next and so on. This had a correlation of -0.0740 with helpfulness ratio.

4.7 Review age

It is possible that the older a review is, the more it's helpfulness ratio. To explore this the correlation of helpfulness ratio with the raw unix timestamp was calculated, which comes out to be -0.1463. It is negative as expected, though quite weak.

4.8 Number of reviews of a product

This might measure how popular or unpopular a product is. There was no correlation found with helpfulness ratio (0.0658)

4.9 Review writing style

Writing style can have many elements, the following were explored:

1. Number of total punctuations
   The correlation was 0.2327, thought the raw number might also be correlated to the length of the text (which is already shown). The correlation when calculated with ratio of punctuation to total characters was quite low at -0.0694

2. Number of title words
   The correlation is 0.2535, though like punctuation the raw number might also be correlated to the length of the text (which is already shown). The correlation when calculated with ratio of title words to total words was quite low at 0.0508

3. Number of words in all caps
   The raw correlation is only 0.026, but taken as a ratio with total words it is -0.1195, which is still quite low but seems to indicate that reviews with more all caps words are voted more unhelpful.

4. Number of exclamation marks and questions marks
   The correlation is quite low at -0.016

5. Average word length
   Average word length has a small correlation at 0.1553

6. Average sentence length
   Average sentence length also has a small correlation at 0.1034

7. Ratio of word misspellings
   This is measured as number of misspelled words divided by total number of words in the text. It was expected that a high ratio might have a negative correlation with helpfulness, but the correlation is only 0.0396 meaning almost no correlation.

8. Readability of text Many studies have shown that readability is an important factor in determining helpfulness. The Automated readability Index (ARI) was calculated for each review and the correlation comes out to be 0.1139. There are other readability scoring techniques, but they were not explored.

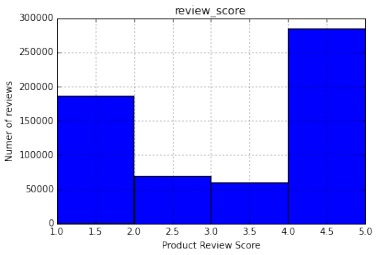

Figure 3: A histogram showing review ratings

Variable	Correlation with helpfulness ratio
Review score	0.5431
Deviation from mean review score (Signed)	0.4688
Deviation from mean review score (Absolute)	-0.1913
Review Length in characters	0.2741
Total number of votes	0.038
Time diff from first review by user	-0.0344
Total number of reviews written by user	0.1513
Time difference from last review (product)	-0.2256
Rank of review in a product by time	-0.0740
Review age	-0.1463
Number of reviews of product	0.0658
Number of total punctuation	0.2327
Ratio of punctuation with characters	-0.0694
Number of title words	0.2535
Ratio of title words with all words	0.0508
Ratio of number of words in all caps with total words	-0.1195
Average words length	0.1034
Ratio of words misspelled with total words	0.0396
Text readability (ARI)	0.1139

Table 1: Summary of correlations with helpfulness ratio

5. PREDICTIVE TASK

5.1 Definition

The task is defined as classification of helpfulness of reviews with two classes: Helpful and not helpful. A review is considered helpful if the helpfulness ratio >=0.6, otherwise the review is considered unhelpful.

where helpfulness ratio is defined as positive votes / total votes for a review.

5.2 Evaluation methods

The evaluation of model can be done by three score types: accuracy, area under the ROC curve and F scores and the

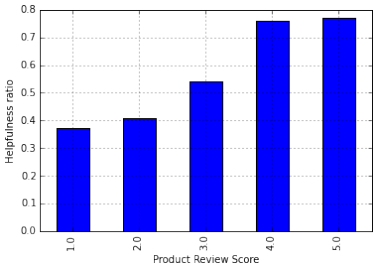

Figure 4: A histogram showing helpfulness ratio for review ratings

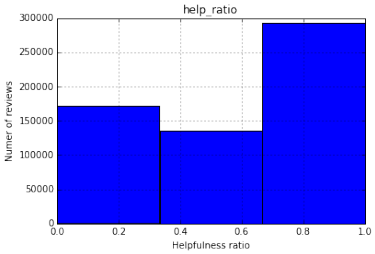

Figure 5: A histogram showing distribution of reviews by helpfulness ratios

final results can be obtained as a mean of scores in 10 folds in a 10 fold cross validation setting, where data points are taken at random in each fold's test and train sets. This will ensure that the results are not just reflecting the scores on a single lucky/ unlucky random split of the data set into training and testing. Accuracy on individual classes can also be obtained.

5.3 Baseline performance

A simple predictor for two classes can be the random predictor which will predict either class at random. That would have an accuracy of 0.5. A better null predictor can always predict reviews as Helpful, and as the mean helpfulness ratio is 0.6, it will have an accuracy of 0.6.

6. PREVIOUS WORK

A lot of work has been done in trying to understand how people feel about opinions of others on a topic or product. Review helpfulness is more of a question of asking what the opinion of a person is for another person's opinion for a product rather than the opinion about the product itself. There have been different approaches to this problem on a variety of datasets. For example [1] tries to model helpfulness ratio (defined in the same way as hepfulness votes / total votes) through previous work on sociology and statistics and prove for a amazon book review dataset that helpfulness is affected independently by factors other than just the textual quality of reviews by using plagiarism of reviews accross products. They showed that the same text (plagiarised reviews) across different products got different helpfulness ratios meaning that factors other than just the text quality are at work.

In [6] the author tries to understand how linguistic features affect helpfulness ratio deriving on the Linguistic Category Model (LCM) as described in [3]. The LCM uses three broad categories: Adjectives, State verbs and action verbs. High use of adjectives like 'fantastic' makes a review more abstract while words like 'love' convey emotion while action words like 'take', 'make' make a review more objective. The author shows that linguistic features are more helpful in predicting helpfulness for 'experience goods' like books and music, and that more objective reviews are rated more helpful (for experience goods)

In [4] the authors try to use SVM regression to rank amazon reviews by helpfulness for mp3 players and digital cameras. They report that review length, unigram features (tf-idf) and product ratings are their most important features. Authors in [5], on the other hand prove that readability and stylistic features are more important than review length (even though they are correlated) on a dataset of amazon UK book reviews.

Liu et. al in [8] attempt to detect reviews with low quality by first removing biases like imbalance vote bias, winner circle bias and early bird bias. They argue that since other studies did not remove these biases their results are subject to them and do not model a review's helpfulness in its pure form. They use amazon review dataset for digital cameras and made their own specification of what properties a good (and conversely bad) review should have. Then they used manual annotation of reviews according to the specifications to avoid aforementioned biases.

Lee and Choeh in [7] develop a back-propagation multilayer perceptron (BPN) model to predict the level of review helpfulness, and show that their results are better than linear regression analysis in terms of mean squared error. The determinants they considered were product data like type, price, Amazon.com sales rank, number of reviews and average rating over time and similarly review extremity and other stylistic features. They confirm that both textual features and product metadata are important in predicting helpfulness.

Ghose and Ipeirotis in [2] try to predict review helpfulness in a very similar fashion on amazon review dataset of three product categories : audio and video players, digital cameras and DVDs. They also used a large number of reviewer, product and review metadata and historical features like reviewer characteristics including hobbies, nicknames and past reviews' helpfulness, product retail price sales rank, average rating, elapsed date, etc. Also, they report that Random forest classifiers worked better for them than SVMs, which were used by other studies.

7. FEATURES

In data exploration we saw many variables in the dataset with varying correlation with helpfulness ratio (and hence varying predictive powers). For example it was seen that a product's total number of reviews has very little correlation with review helpfulness and similarly writing style elements like punctuation use and misspellings have no correlation either. For the preliminary set of features those variables were selected which had at least 0.1 |correlation| with helpfulness. It is possible that many of these variables are highly correlated themselves. A good feature set would have features with good correlation with helpfulness ratio but low

correlation between themselves. A preliminary set of features by eliminating variables found with low correlation in data exploration stage are given below:

1. Review product rating
   (RW_SCORE)
   The intuition behind this feature comes from the results we saw in data exploration section, where very highly or extremely lowly rated products often have the most helpful reviews.

2. Review text length
   (RW_LEN)
   Longer review texts can be perceived to be more thorough and many previous studies have shown that people mark such reviews as more helpful.

3. Total reviews written by user
   (USR_N_RW)
   As explained previously, this might reflect the interest a user has in movies or review writing. It had a weak correlation with helpfulness ratio as seen earlier.

4. Time delay of a review
   (RW_TM)
   Due to the early bird bias reviews appearing first might be voted as being more helpful, as they might be seen my much more number of people. Though this is not a completely accurate statement, as we saw that more number of votes does not correlate with more helpfulness ratio.

5. Age of a review
   (RW_AGE)
   Older reviews in general were shown to have some correlation with helpfulness ratio, such that older reviews were found ot be slightly more helpful.

6. All capitalized words ratio (CAPS)
   Capitalized words have always had a special significance in the online community. They are associated with the equivalent of shouting in text and used to convey strong emotions toward something. Very often, they are considered rude. This is reflected in a small negative correlation we found earlier.

7. Average sentence length
   (SENT_LEN)
   Sentence length can have a great impact on comprehensibility of text, and easily readable text should appeal to more people. It was found ot have a small correlation with helpfulness ratio earlier, so we can include it.

8. Average word length (WORD_LEN)
   A higher average word length may mean use of more complicated words and impact readability in some way.

9. Question and exclamation marks ratio
   (SURP)
   A lot of exclamation marks and question marks can make the review much more personal or subjective ('Loved it!', 'BORING!'). This is calculated as a ratio with total characters.

10. Review score deviation from mean-
    Signed (SCR_DEV_S)
    This is in accordance with the conformity principle as explained before that people may find reviews conforming to the average more helpful.

11. Review score deviation from mean-
    Absolute (SCR_DEV_ABS)
    Similar to SCR_DEV_S, but it ignores the direction of the deviation. It was found to have a negative correlation which makes sense since people generally agree with the average opinion.

12. Automated readability index (ARI)
    This is a simple metric of readability and was found to have a slight correlation with helpfulness.

13. Term frequency inverse document frequency (TFIDF)
    This feature was added to include unigram features for text. The high dimensional result is passed through truncated Single Value Decomposition (the process is called Latent semantic analysis) with resulting 100 dimensions. This way we model semantic patterns between different words in the review text.

The scatter matrix in Figure 6 visualizes the correlation between each of the features given above. It can be seen that most features are not visibly correlated between themselves. Automated readability index is highly correlated with average sentence length, which is interesting as it means the ARI considers sentence length as an important factor in readability. Thus we will take average sentence length (SENT_LEN) as the feature over ARI as it was shown to have higher correlation with helpfulness ratio than ARI. Similarly SCR_DEV_A and SCR_DEV_ABS have an expected relationship since they are just the signed and abslute value version of the same quantity - deviation from mean. We will take SCR_DEV_S as a feature over SCR_DEV_ABS as it had higher correlation.

Review length and score are also somewhat correlated as can be expected since we saw earlier that better reviewed products tend to have longer reviews as well. Use of surprise characters (! and ?) also follows the review ratings' pattern closely and it can be seen that it is highest for reviews with either a 1 star rating or a 5 star rating. Hence the SURP feature is also discarded.

The special dictionary feature (SP_DICT) seems to be correlated with review scores as well, as people are using more 'movie related words' when they are writing a 1 star review or a 5 star review. Hence this feature is also discarded.

All features are normalized to 0 mean and unit variance, as classifiers like SVM with RBF kernel expect the features to be normalized this way.

8. MODEL DESCRIPTION

8.1 Overview

The scikit learn tool in python gives a large variety of classifiers like Logistic regression, SVM (with RBF, polynomial or linear kernel), decision trees, ridge classifier and then many ensemble based classifiers like Bagging classifier, Extra trees

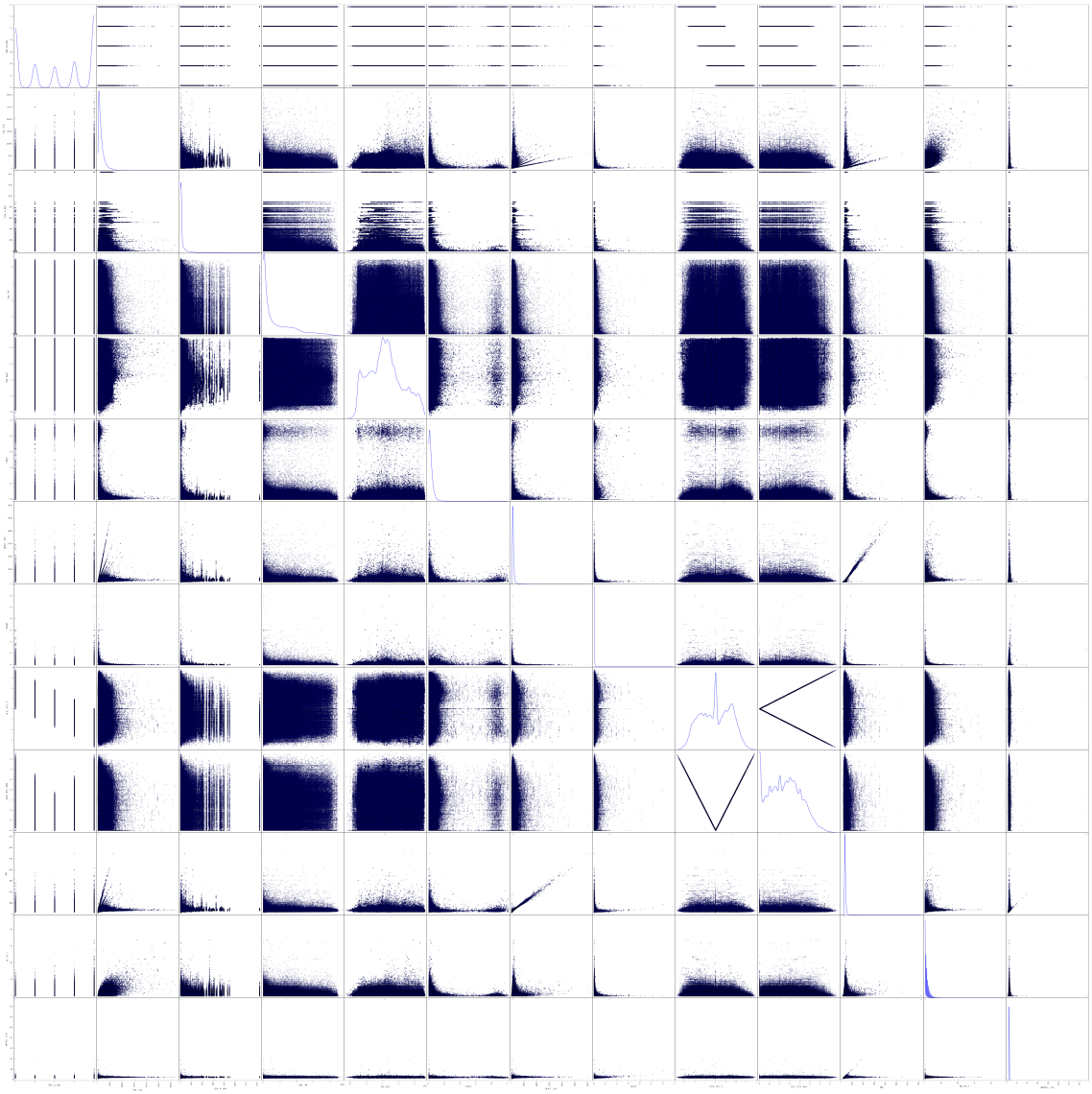

Figure 6: Scatter matrix for features given in section 7. Diagonal is Kernel density estimation plots

classifier, Random forest classifier, Passive Aggressive classifier. It is a time consuming and non-trivial, but important step to determine what the best model for the task is. Here the methodology used is to try all these classifiers in a grid search fashion to see which performs the best with the feature set.

8.2 Comparison of models

It turned out that ensemble tree based methods were best performing out of all the available methods. SVM with any kernel failed to converge even after a long time. Tree based classification like ExtraTrees classifiers and RandomForestClassifiers far outperformed SVM (linear, polynomial or RBF kernel) and Logistic Regression. Another ensemble based classifier, the Bagging classifier was close behind, while other classifiers like SGD classifier, Passive Aggressive classifier and Ridge classifier were far behind with accuracies in low seventies.

Between Extra trees and Random Forest, Random Forest was chosen since it had better performance in a 10 fold cross validated setting. Extra tress was overfitting more than random forest as it slightly outperformed it in a random but static 60:40 train:test split, but lagged behind in cross validation mean across folds.

9. RESULTS

9.1 Scores

The best score was achieved with a combination of RW_SCORE, TEXT_F, TFIDF features using the Random Forest Classifier with parameters given in section 9.3. The mean accuracy of 10 folds in 10 fold cross validation comes out to be 85.95%, while on a random split of 60:40 train:test the accuracy, AUC and F score are 90.80, 90.69 and 91.00 respectively.

These scores are somewhat better than others who have attempted classification in the same way on similar data sets. Ghose et. al in [2] for example report 78.79% accuracy and 0.73 AUC on amazon DVD reviews on a similarly sized data sets. They did not consider review rating as a feature but tried to include subjectivity and reviewer metadata as features. Comparison with other works is harder due to difference in the kind of datasets (mostly book and electronic products reviews) and their size. The dataset size considered here, which is 600,000 reviews compares favorably with most other studies which have had smaller datasets (<50,000).

Feature Combination	Accuracy	AUC	F1 score
RW_SCORE	76.71	79.18	75.85
SCR_DEV_S	68.51	71.09	66.97
TEXT_F	67.19	74.78	69.38
TFIDF	71.59	79.55	73.04
TIME_DIFF	51.61	49.94	56.89
RW_SCORE+TEXT_F	81.35	88.66	82.23
RW_SCORE+TFIDF	81.35	88.60	82.52
RW_SCORE+TEXT_F +TFIDF	85.95	92.79	86.92
RW_SCORE+TEXT_F +TFIDF+USR_N_RW	81.00	87.94	81.98
RW_SCORE+TEXT_F +TFIDF+RW_AGE	80.88	87.97	82.13
RW_SCORE+TEXT_F +TFIDF+TIME_DIFF	80.68	87.79	82.09

Table 2: Results of classification with different feature combinations averaged over 10 folds of cross validation

9.2 Most useful features

Results from all feature combinations using mean of 10 folds in 10 fold cross validation can be seen in Table 2. As it can be seen the most useful features turned out to be review score, text style based features and unigram features (tfidf). This is in line with what other have reported previously.
It is somewhat surprising that features targeted towards biases in reviews like the TIME_DIFF feature (targeted towards the early bird bias) and deviation from mean features like SCR_DEV_S did not contribute to scores as much as expected and as much as reported by some of the studies [1]. It appears that biases do not contribute as much to predictive power of helpfulness in this data set as authors in [8] anticipated where they invested a lot of effort in removing these biases like manual annotation. Interestingly, the hypothesis that more experienced users will write better reviews could not be proven with these results as such features failed to improve the scores. Similarly features like product popularity failed to have an impact on prediction.

9.3 Model parameter tuning

Parameters for the Random Forest classifiers are n_estimators, which is how many trees should be used in total, max_features which is the number of features when considering the best split, max_depth which specifies what the maximum depth of any tree should be and bootstrap which is a flag indicating if bootstrap samples should be used when building trees. These parameters were tuned using a grid search accross values of n_estimators as 5, 10, 20,30,40,50 for max_features as 'auto', 'sqrt', 'log2' and None, max_depth as 5,50,100,150 and bootstrap as true and false.

The optimal parameters found by grid search on 10 fold cross validation were: n_estimators = 50, max_depth=None (unlimited depth), max_features = 'log' and bootstrap = True.

10. CONCLUSION

A model of predicting helpfulness in reviews was developed with an accuracy of 86% on Amazon movie reviews dataset. The most important features were review length and other writing style features like sentence length, words length use of punctuations, etc. and unigram features using TF-IDF followed by Truncated SVD. Features targeting biases to improve score like review time based features (for early bird bias) and deviation from mean (for conformity principle) proved to be not useful in prediction as others have reported previously. Features like user experience measured in different ways or product popularity also failed to have a positive effect on prediction.

11. REFERENCES

[1] C. Danescu-Niculescu-Mizil, G. Kossinets, and J. Kleinberg. How opinions are received by online communities: A case study on amazon.com helpfulness votes. *Proceedings of WWW*, 2009.

[2] A. Ghose and P. G. Ipeirotis. Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics. *IEEE Transactions on Knowledge & Data Engineering*, 2010.

[3] S. GR and K. Fiedler. The linguistic category model, its bases, applications and range. *European Review of Social Psychology*, 1991.

[4] S.-M. Kim, P. Pantel, T. Chklovski, and M. Pennacchiotti. Automatically assessing review helpfulness. *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, 2006.

[5] N. Korfiatis, E. GarcÃ■a-Bariocanal, and S. SÃąnchez-Alonso. Evaluating content quality and helpfulness of online product reviews: The interplay of review helpfulness vs. review content. *Electronic Commerce Research and Applications*, 2011.

[6] S. Krishnamoorthy. Linguistic features for review helpfulness prediction. *Expert Systems with Applications*, 2015.

[7] S. Lee and J. Y. Choeh. Predicting the helpfulness of online reviews using multilayer perceptron neural networks. *Expert Systems with Applications*, 2014.

[8] J. Liu, Y. Cao, C.-Y. Lin, Y. Huang, and M. Zhou. Low-quality product review detection in opinion summarization. *Association for Computational Linguistics*, 2007.

[9] Y. Liu, X. Huang, A. An, and X. Yu. Modeling and predicting the helpfulness of online reviews. *IEEE international conference on data mining*, 2008.

[10] J. McAuley and J. Leskovec. From amateurs to connoisseurs: Modeling the evolution of user expertise through online reviews. *International World Wide Web Conference Committee (IW3C2).*, 2013.