

What makes a board game highly rated?

Student: Francis Semenuk –
500185004

Supervisor: Ceni Babaoglu

Submitted: June 6th 2022



Table of Contents

Abstract.....	3
Literature Review	4
Methodology	7
Univariate Analysis	8
Final Data table	11
Model methods	12
Overall Methodology	13
Results	14
Decision Tree Classifier	14
Dimension reduction using PCA.....	16
Decision Tree Classifier Final Results.....	17
K-nn Classifier.....	19
Naïve Bayes Classifier	21
Model Comparison.....	23
Discussion and Conclusion	25
References.....	27

Abstract

Game designers want their boardgames it to be a smash hit that is highly rated that many people will play and acclaim its glory. To determine if a boardgame will be well rated, and therefore successful, one could start by analysing the structure of the boardgames to discover what kind of structure would be more likely to ensure high ratings.

This study will focus on trying to find out patterns within data of boardgames and try to determine by machine learning, given a new set of board game data, where the game may fit for an overall rating. Data used in this study is available from Kaggle.com and is an SQLite dataset (<https://www.kaggle.com/datasets/gabrio/board-games-dataset>). Boardgames will attempt to be classified as highly rated (7.0 or greater for the stats.average attribute), mediocre (5.0 to 6.9) or low (less than 5.0) based on the criteria provided. The main table dataset that is 81 attributes deep with over 90,000 records will be used as the focus. Various tools will be employed to explore the question of highly rated board games, including: SQLite to initially explore the data, reducing the data down to boardgames and the number of expansions related to it, along with most of the attributes intact; Python will then be used to explore the data further, using linear regression algorithms to assist in dimension reduction; Python to also analyze the dataset using classification tree algorithms (using 10-fold training) and k-nearest neighbors, furthermore using Naïve bayes algorithms to assist in classification based on the genres each method being compared to achieve strongest results; Python will also be used for data visualizations; and Tableau will be used for additional data visualizations not achieved while using Python.

The intended outcome will allow for classification of a new boardgame, based on the general data, as to if it will likely be highly rated (2), mediocre (1) or low rated (0) for the stats.average dependant variable.

Literature Review

Board games have been around for thousands of years, some boardgames, such as knucklebones, dating back to the time of the ancient Egyptians. More recently, in the last century, boardgames started to take a hold of the market and have seen steady gains. Not until recently, about the last decade, boardgames have started to see a precipitous climb in the amount of users (Roeder 2015), sales and profit in no small part due to advances in globalization, digital technology and trade.

Let me paint a picture on the canvas of what is known about boardgame data and its sources. Boardgame data is available with a high volume of collected data, over 90 thousand entries, in the BoardGameGeek (BGG) database. Much of the data contained is 'demographic' data of various games, such as number of players, playing time, game descriptions, genres and more, which comes from boardgame developers. In addition, there is also a large community or social involvement which allows users to rate, rank and comment on boardgames in areas such as; suggested age, recommended number of players, people who own the game and more, which can add an interesting layer of how well received a board game may be by the community as a whole. It is also known that since the inception of crowdfunding platforms, such as Kickstarter (Jae-Won et al. 2020), sales and volume of boardgames have seen great increases, necessitating a need to understand the market in different ways. Comments through BGG have been made about some game 'failures', such as the theme or genre not really matching well to the game mechanics or seen as an afterthought surrounding the mechanism of the game rather than well integrated. Meaning that theme can have an important role in the selection and preferences of games over another. It is unclear how much this factors into the overall rating of a game and user comments were not available for thorough study in this analysis.

When investigating into literature about studies on regression analysis or machine learning, very few studies were found despite the availability of data. Specifically, no article could be found investigating whether there were any links between the user rating of games and the game demographics. Likely, larger businesses would have done a market analysis based on sales and generated

projections based on the criteria of sold copies and any focus groups or game testing. However, the statement on the aforementioned businesses is purely speculative.

Even though there are gaps in studies on features which create best user experiences, there is a plethora of related information pertaining to reviews ratings or purchases and predictions based on those. Most studies found and reviewed extended to evaluating the comments or user experience and trying to predict a specific outcome. According to Siersdorfer et al. (2010) they found specific categories attract different users which will generate more or less discussion as function of the controversy of their topics. They did this by studying the influence of sentiment by using the SentiWordNet thesaurus. Similarly, Turney (2002) applied Pointwise Mutual Information and Information Retrieval to find that predicted sentiment of a review could be accurately obtained, depending on the topic, up to 84% of the time. Khan et al. (2020) also showed that predicted movie genre preferences could be discovered through decision trees to an accuracy of 68.5%, this was done through analysis of personality and values. Guo *et al.* (2017) also analyzed review data using Latent Dirichlet allocation for topic modelling after reviews were processed and parts of speech tagged and using regression analysis. Going a step beyond the reviews to discover if reviews are rated as helpful or not, Danescu-Niculescu-Mizil *et al.* (2009) found that a review's perceived helpfulness depended on the content and relation of its score to the other scores. Meaning the reviews were seen as more helpful if closer to the average, doing this by signed deviation. One study (Ghose & Ipeirotis 2007) went a step further and attempted to predict the usefulness of review using text mining techniques and subjectivity analysis, they were able to find the most helpful reviews for users and display them first. Game recommender systems have also been derived for Board-game platforms (Jae-Won et al. 2020) and video game platforms (Shahian et al. 2020) and on appealing items based on previous data (Steck 2013), however these recommender systems fall short of creating recommendations for game designers on what they should develop.

This study will analyze the underlying structure of boardgames to derive any meaningful features that should be focused on by boardgame designers. This will allow for a feedback loop of users informing the designers of what is desired and the market

involved. This is akin to the recommender systems, however with the lack of comment data analysis will be aimed at the features of the games instead.

The boardgame community is a small (comparatively to the video game industry) but fiercely loyal community in which feedback is essential to the life of any boardgame at its core. As this community continues to gain momentum it becomes even more important for novice and artisanal board game designers navigate the most common or preferred user experience. This can inform designers on how to increase sales and output based on this user feedback.

Methodology

The dataset was explored to find that some of the board games listed in the game.type column contained both games as well as expansions. Since only boardgames were of interest, the data was flattened, through SQL to incorporate expansions as an additional field as a count.

After consecutive analysis and cleaning of columns with high amount of nulls, reduction by removing highly correlated columns and eliminating columns that had excessive amounts of categories (over 1000) there were 26 columns remaining that were used for the final data modeling.

```
# Column
---
0 details.maxplayers
1 details.minage
2 details.minplayers
3 details.playingtime
4 details.yearpublished
5 attributes.total
6 stats.average
7 stats.averageweight
8 stats.bayesaverage
9 stats.stddev
10 stats.trading
11 stats.usersrated
12 stats.wanting
13 polls.language_dependence
14 polls.suggested_numplayers.1
15 polls.suggested_numplayers.10
16 polls.suggested_numplayers.2
17 polls.suggested_numplayers.3
18 polls.suggested_numplayers.4
19 polls.suggested_numplayers.5
20 polls.suggested_numplayers.6
21 polls.suggested_numplayers.7
22 polls.suggested_numplayers.8
23 polls.suggested_numplayers.9
24 polls.suggested_numplayers.Over
25 polls.suggested_playerage
26 expansions
```

Figure 1. Final listing of the 26 columns from the initial 82 dimensions in the dataset.

After clearing of columns with high amounts of nulls (>80% in columns that could not otherwise be imputed) a univariate analysis of each of the numeric columns revealed many columns which had data that varied quite widely.

Univariate Analysis

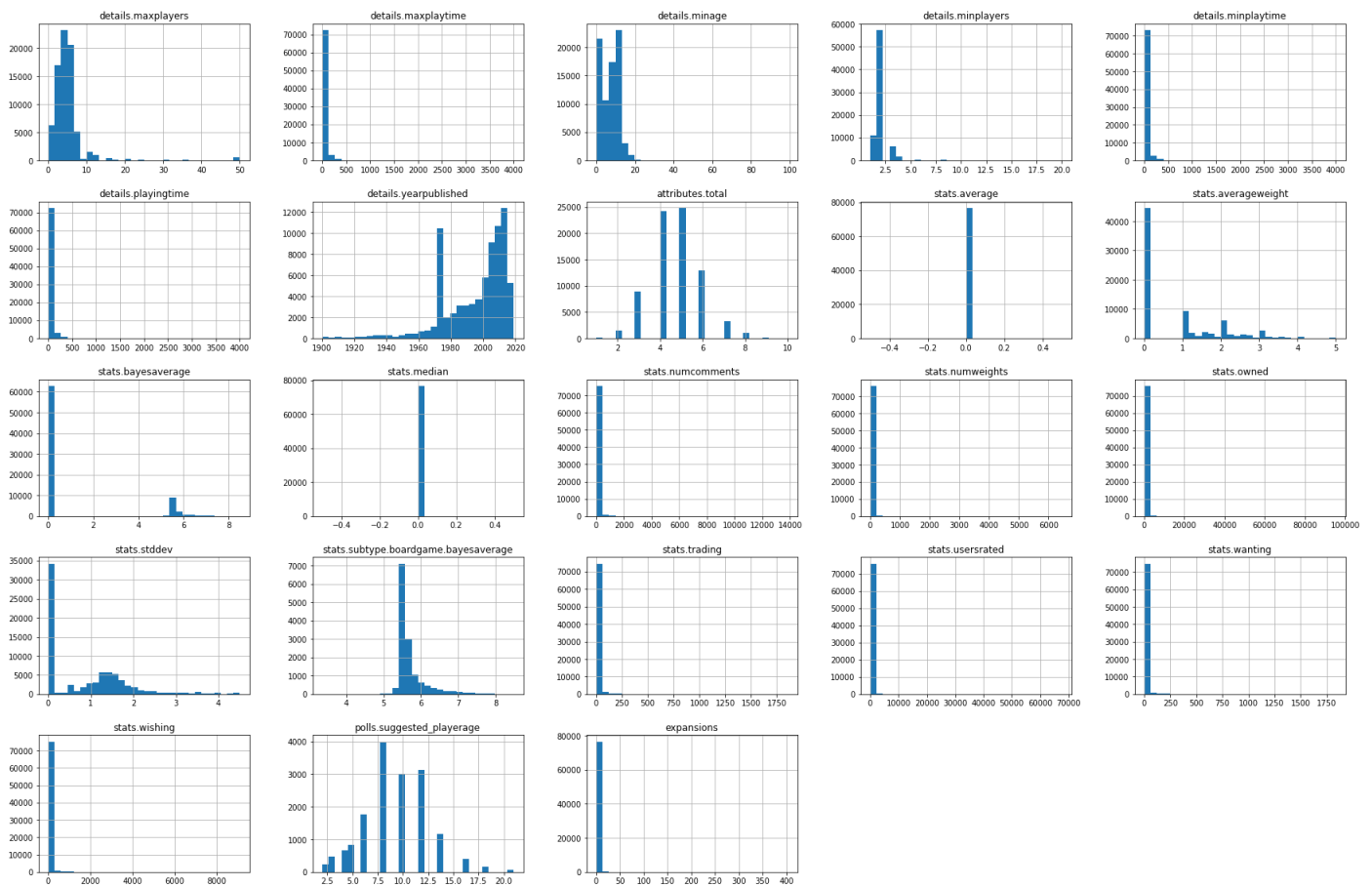


Figure 2. Histograms of categorical and numeric data.

The categories of average ratings appear to be well represented (fig.1) where each category has over 10,000 data points in each. Some of the histograms could be better represented if further outliers were removed. In the bivariate analysis, there are several highly correlated columns which will allow for one of each set to be removed (e.g. details.playingtime, stats.numweights, stats.owned, stats.subtype.boardgame.bayesaverage, stats.wishing).

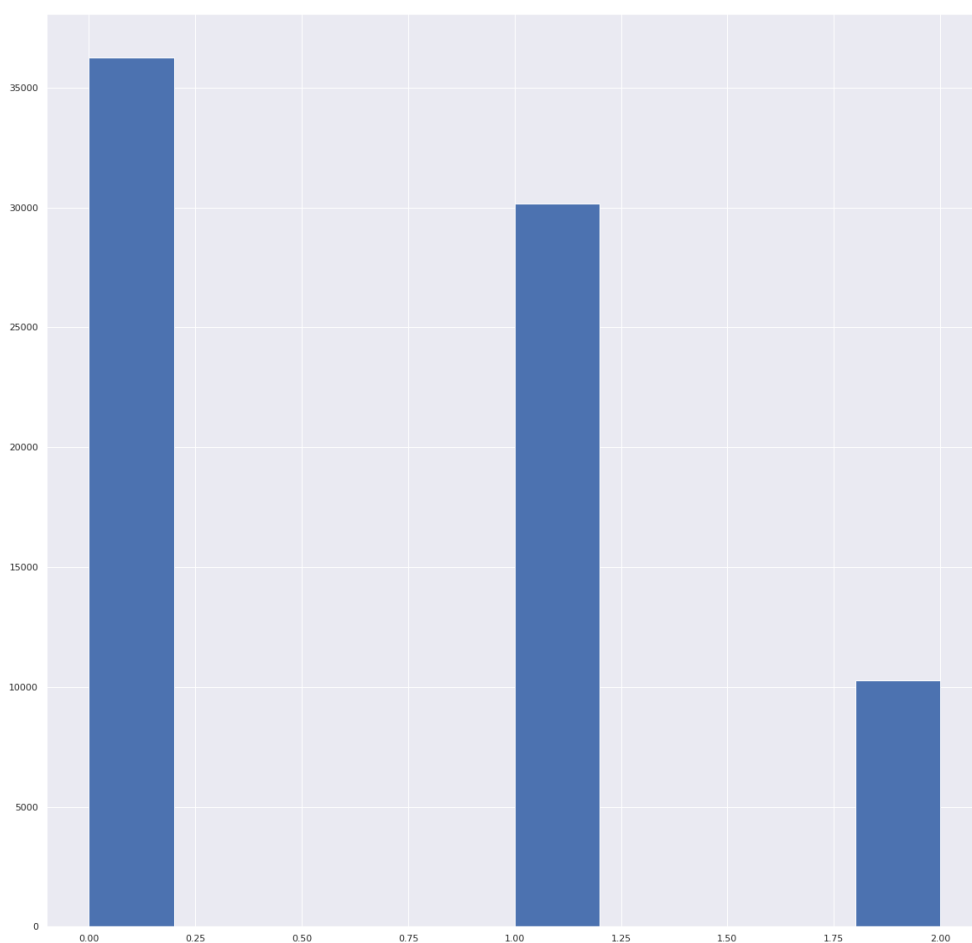


Figure 3. Categories of represented stats.average, left to right; low, medium, and highly rated

Bivariate analysis revealed several highly correlated columns (eg. Details.maxplaytime and Details.playingtime). One of each set were removed to reduce the dimensions of the dataset.

Bivariate Analysis



Figure 4. Bivariate analysis of numeric columns showing correlation of individual columns

Final Data table

	maxplayers	maxplaytime	minage	...	numplayers.8	numplayers.9	numplayers.Over	suggested_playerage	expansions
1	5.0	240.0	14.0		NotRecommended	NotRecommended	NotRecommended	14.000000	0
2	4.0	30.0	12.0		NotRecommended	NotRecommended	NotRecommended	9.187473	0
3	4.0	60.0	10.0		NotRecommended	NotRecommended	NotRecommended	10.000000	0
4	4.0	60.0	12.0		NotRecommended	NotRecommended	NotRecommended	14.000000	0
5	6.0	90.0	12.0		NotRecommended	NotRecommended	NotRecommended	12.000000	0

Table 1. Header rows of the first 5 rows in the dataset. Preceding names before the '.' were removed to allow for room on the page. There are a total of 26 remaining attributes and 76688 rows

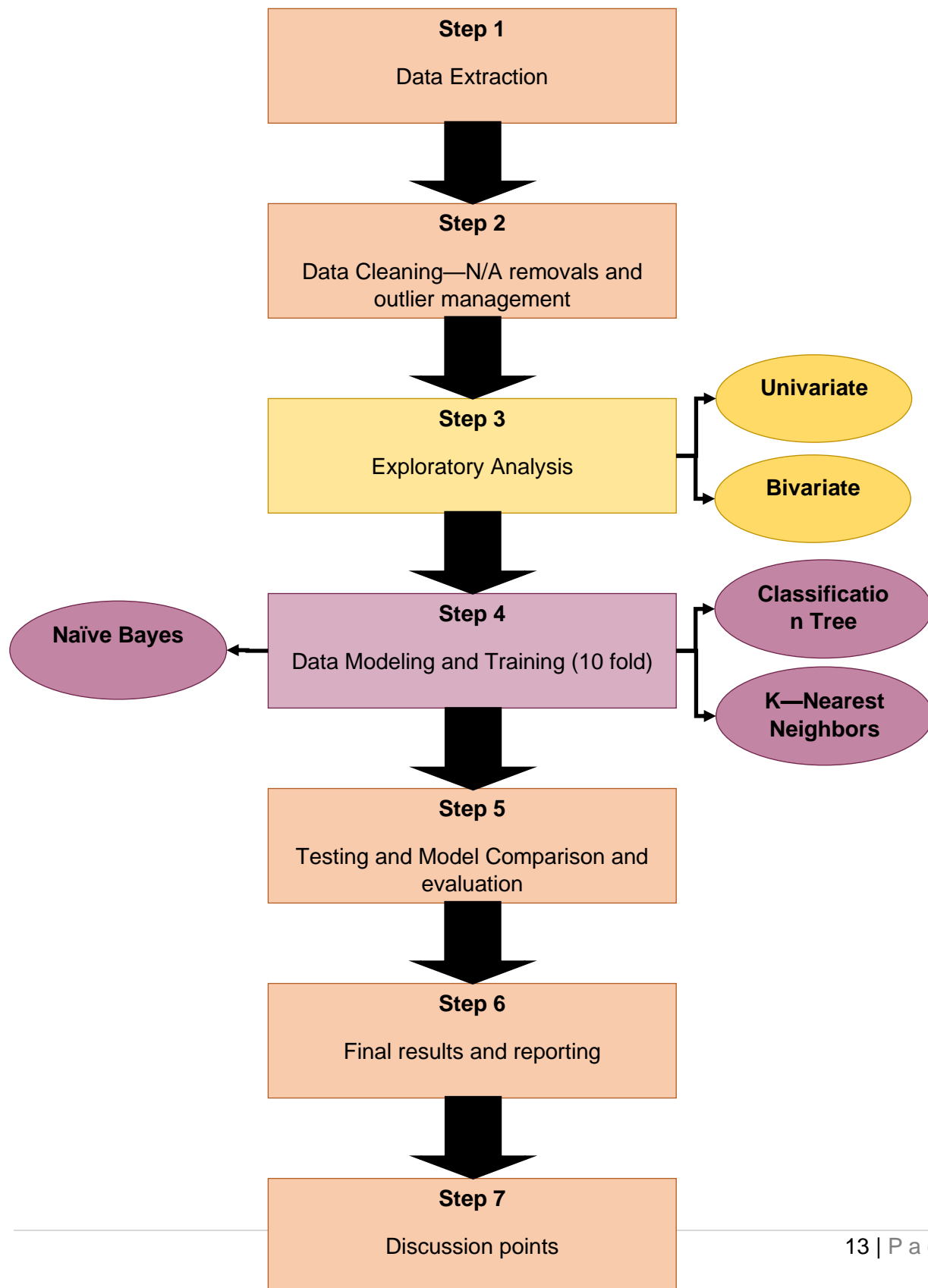
Model methods

For the analysis of the boardgame dataset, three machine learning algorithms were chosen, based on the ease of use and implementation: decision tree classifier, naïve bayes classifier and k-nearest neighbors classifier. The decision tree and knn classifiers were attempted to be tuned to adjust to the most ideal settings for the models.

After the first model is analyzed, the set will further be compared using principle component analysis to observe whether this will improve the overall performance of models.

Once all the models are prepared, they will each be assessed against the same k-fold of the dataset and the results will be assessed. Further to the assessment, ANOVA and a Tukey's posthoc tests will be run to ensure check if the resultant sets are significant from one another.

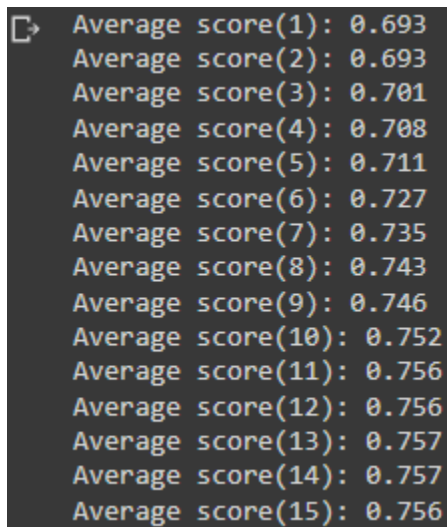
Overall Methodology



Results

Decision Tree Classifier

The first run model was a decision tree classifier. After splitting the dataset into the relevant folds, and one-hot encoding non-numeric data, the model was tuned to find out the most ideal outcomes for the number of leaves.



```

Average score(1): 0.693
Average score(2): 0.693
Average score(3): 0.701
Average score(4): 0.708
Average score(5): 0.711
Average score(6): 0.727
Average score(7): 0.735
Average score(8): 0.743
Average score(9): 0.746
Average score(10): 0.752
Average score(11): 0.756
Average score(12): 0.756
Average score(13): 0.757
Average score(14): 0.757
Average score(15): 0.756

```

Figure 6. Tuning of the decision tree classifier. The average score was produced after the number of leaves that were used to determine the results.

As the above results suggest, a tuning of 12 to 13 leaves creates the optimal output score. This resulted in an output score of 0.75 which may be a good fit for the model. Later on, the decision tree classification model is analyzed using the confusion matrix.

Having a model of 13 leaves is difficult to display, so for simplicity the first 5 leaves are produced in the next figure.

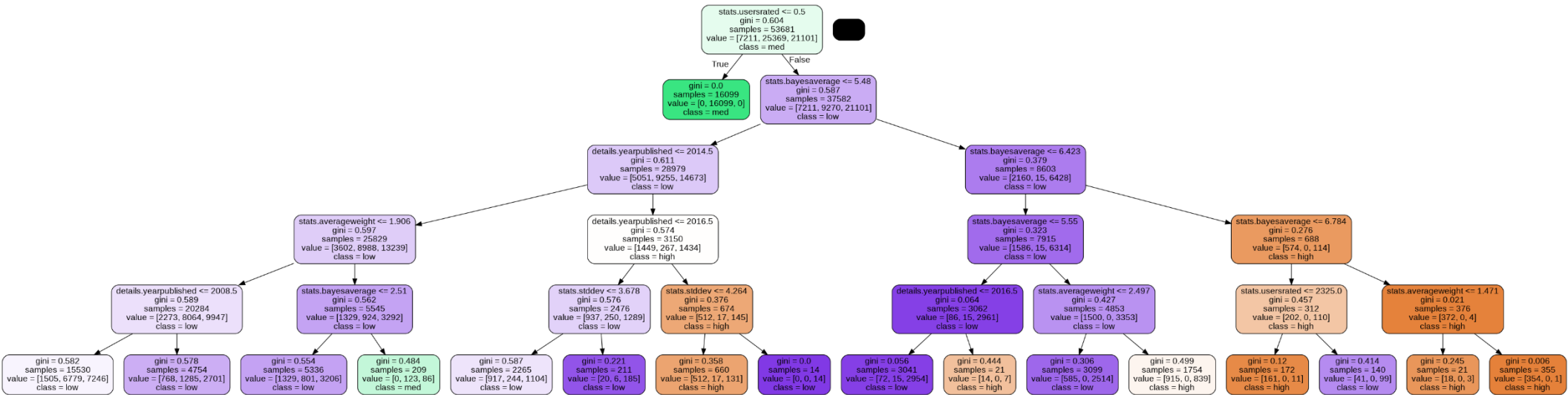


Figure 7. Resultant decision tree with 5 leaves

From the figure 7, it can be seen that the initial leaf of stats.usererrated <=0.5 provides results for the largest proportion of medium rated games. Subsequently, stats.bayesaverage and details.yearpublished produce the next most meaningful results.

Dimension reduction using PCA

The dimensions of the model was reduced using PCA to check if there would be possible improvements to the model.

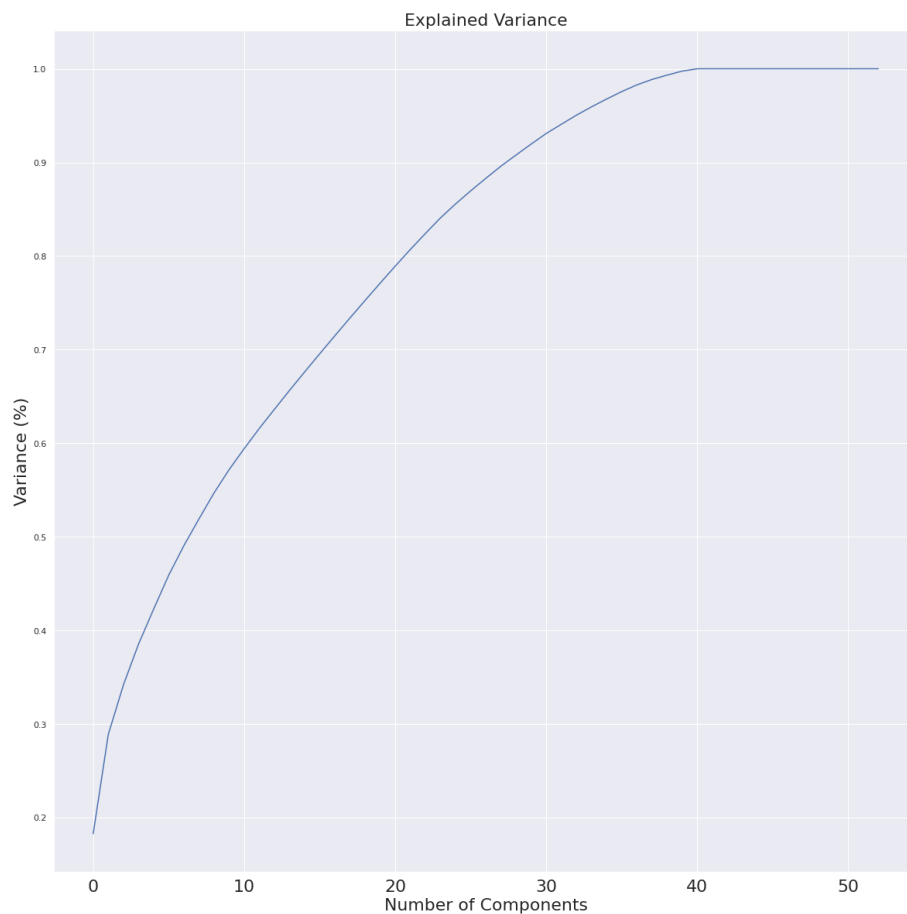


Figure 8. Dimension reduction using PCA, with 95% of the explained variance from approximately 30 components

Using approximately 30 dimensions with PCA resulted in an overall reduction in performance of the model, dropping to ~0.668. In this case PCA would not be worthwhile using for feature reduction, possible due in part because of many of the columns being generated from earlier splits using the one hot encoding.

Decision Tree Classifier Final Results

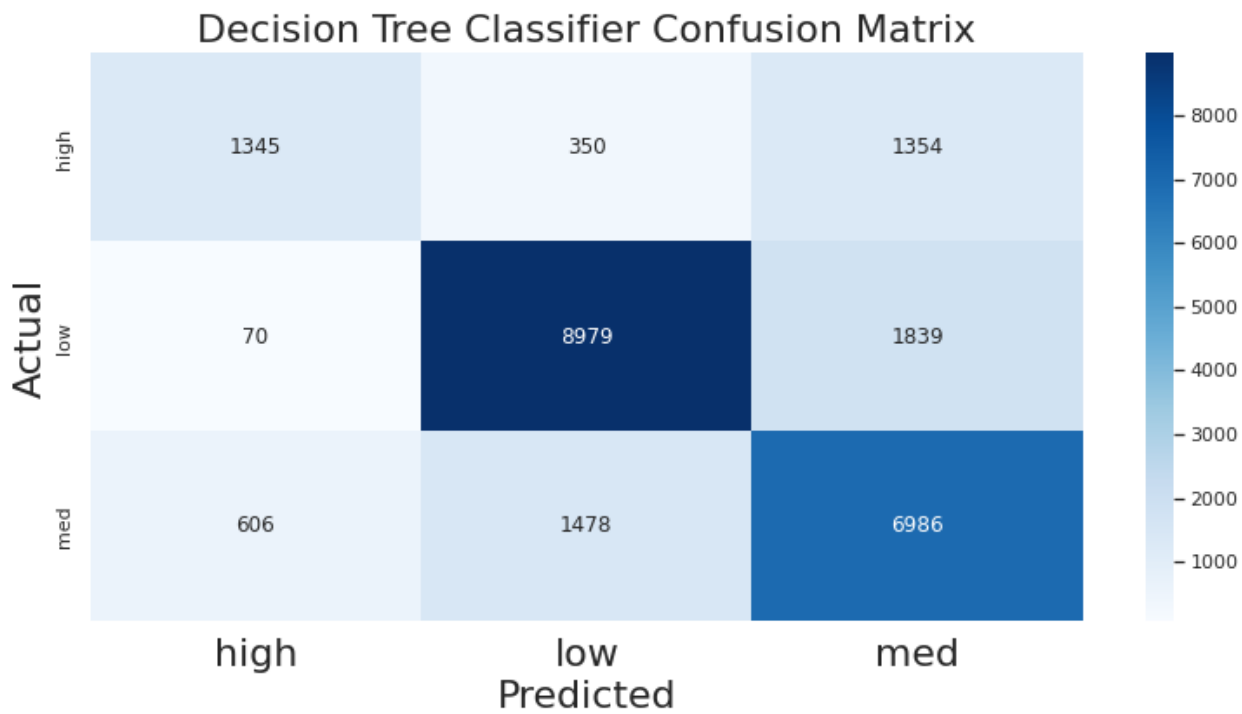


Figure 9. Decision tree classifier confusion matrix representing predicted and actual values for low medium and high categories of stats.average

Table 2. Decision tree classifier confusion matrix calculation results

	precision	recall	f1-score	support
high	0.67	0.44	0.53	3049
low	0.83	0.82	0.83	10888
med	0.69	0.77	0.73	9070
accuracy			0.75	23007
macro avg	0.73	0.68	0.69	23007
weighted avg	0.75	0.75	0.75	23007

While the overall model for the decision tree classifier had a very good accuracy of 0.75, the overall precision also was on the high side for each category (0.67, 0.69, and 0.83 for High, Medium, and Low respectively). An overall weighted f-score of 0.75 also indicates a very good fit for this model. It should be noted, however that the high group was incorrectly classified as medium approximately half of the time it would seem that the model is better at identifying those board games which would be rated at Low or higher. Looking at the individual features from the decision tree - the most important features are the stats.bayesaverage and then the year of publication.

After 10-fold stratified analysis using the Decision Tree Classification model, an average score of 0.756 was achieved with a narrow range of variation of 0.01695136 between the minimum and maximum scores, suggesting high reproducibility of the model.

K-nn Classifier

The second model ran was a k nearest neighbor algorithm. Using the same splits and folds as the decision tree classifier, the following confusion matrix was produced.

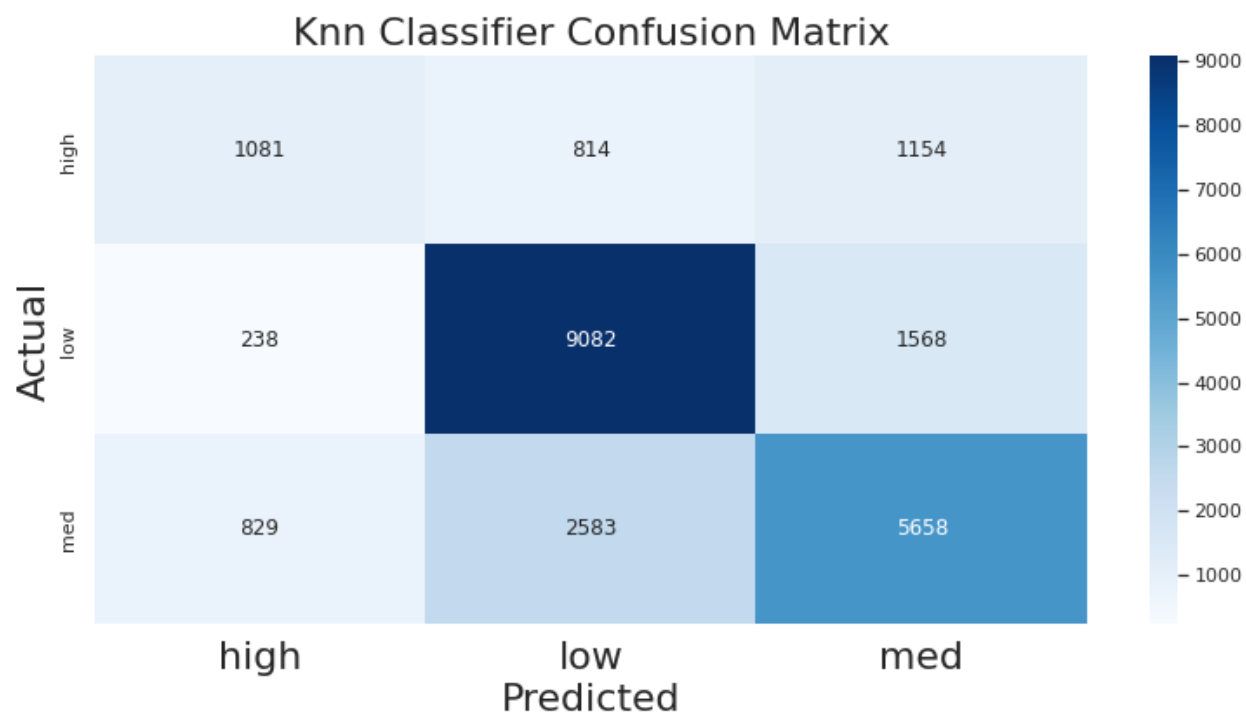


Figure 10. K-nn classifier confusion matrix representing predicted and actual values for low medium and high categories of stats.average

Table 3. K-nn classifier confusion matrix results

	precision	recall	f1-score	support
high	0.50	0.35	0.42	3049
low	0.73	0.83	0.78	10888
med	0.68	0.62	0.65	9070
accuracy			0.69	23007
macro avg	0.64	0.60	0.61	23007
weighted avg	0.68	0.69	0.68	23007

The overall accuracy of the Knn classifier model had a good score of 0.69 with a weighted f-score of 0.68. Most of the accurate precision once again came from the proper classification of the low group (0.73), while precision of the high group was close to that of a coin flip at 0.50. Once again it appears that the models have difficulty in identifying board games that belong in the high or medium category with well matched total values in each.

Analysis of 10-fold stratified splitting with the K-NN classification model revealed an average score of 0.69 with a variation of 0.01816672 between the minimum and maximum scores, suggesting high reproducibility of the model.

The K-NN classifier was attempted to be tuned from 1 to 50, however in choosing a tuning algorithm, the code would hang (>30 minutes) and was prohibitive in producing results. It was then decided to abandon the algorithmic tuning and experimented by individual runs. Choosing to run this way between 5 and 25 neighbors there was little variation in the overall results (hovering between 0.67 to 0.69) and in the end 5 neighbors were chosen as the criteria for expediency of code execution.

Naïve Bayes Classifier

The final model chosen was the Naïve Bayes Classifier. Similar to the decision tree and k-nn models the same splits and folds were used where the following confusion matrix was produced.

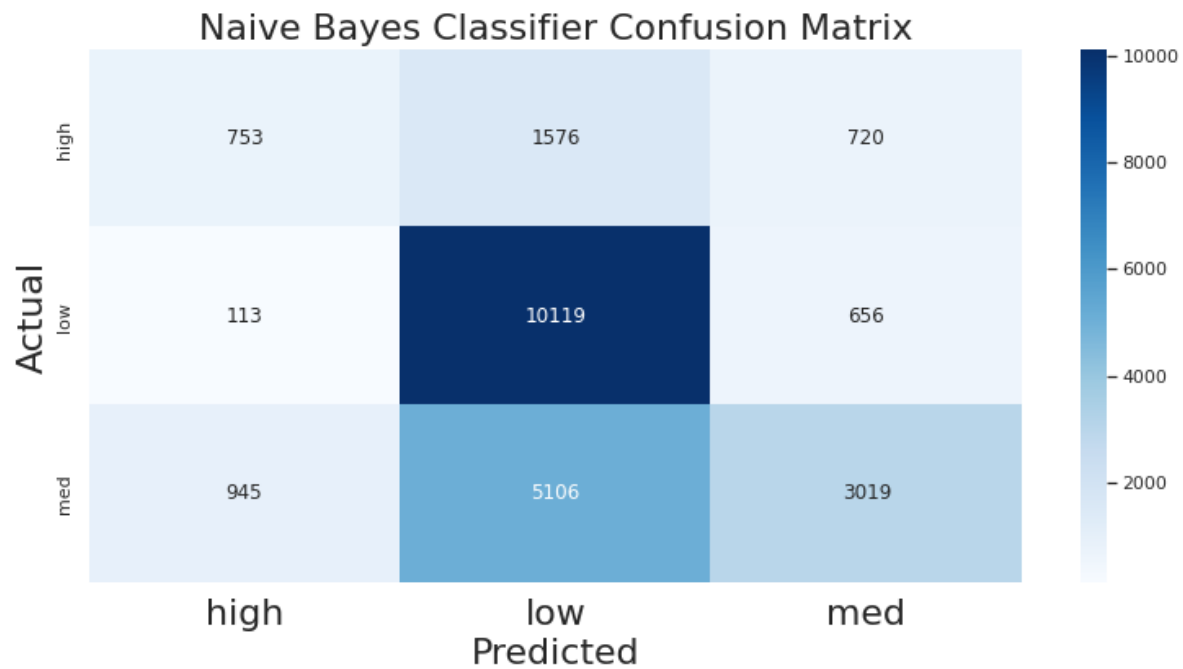


Figure 11. Naïve Bayes classifier confusion matrix representing predicted and actual values for low medium and high categories of stats.average

Table 4. Naïve Bayes classifier confusion matrix results

	precision	recall	f1-score	support
high	0.42	0.25	0.31	3049
low	0.60	0.93	0.73	10888
med	0.69	0.33	0.45	9070
accuracy			0.60	23007
macro avg	0.57	0.50	0.50	23007
weighted avg	0.61	0.60	0.56	23007

The Naïve Bayes model had a fairly low accuracy rating of 0.60. Overall precision also was 0.42, 0.69, and 0.60 for High, Medium, and Low respectively. It would appear that Naïve Bayes was poor at assigning high category games, while best at assigning medium rated games. Similar to the other two models, the Naïve Bayes model also struggled to properly predict highly rated games as those rather than medium rated.

Analysis of 10-fold stratified splitting with the K-NN classification model revealed an average score of 0.61 with a variation of 0.01817771 between the minimum and maximum scores, suggesting high reproducibility of the model.

Model Comparison

Each of the means of the k-fold models were measured and can be seen below:

Model	Score
Decision Tree	0.756285
K Nearest Neighbors	0.689964
Naïve Bayes	0.606040

Plotting the models, using each fold as a measure, together on a boxplot it is evident that there are some differences between the models.



Figure 12. Boxplot comparison of each model used and their k-fold scores.

To ensure that the means were different from one another and that each set was distinct in its measures, analysis of variance was conducted and achieved the following results.

	sum_sq	df	F	PR(>F)
C(group)	0.113385	2.0	1853.569256	1.255568e-29
Residual	0.000826	27.0	NaN	NaN

The degrees of freedom of the model groups was 2 with an F-score of 1853.569256 and a p-score of 1.255568e-29. Since the p-score was much smaller than 0.05 the null hypothesis can be rejected and that there at least one of the data models are significantly different from the others.

To find out which of the models differs significantly a Tukey post hoc test was conducted. The post hoc test achieved the following results:

Multiple Comparison of Means - Tukey HSD, FWER=0.05						
group1	group2	meandiff	p-adj	lower	upper	reject
DT	KNN	-0.0663	0.001	-0.0725	-0.0602	True
DT	NB	-0.1502	0.001	-0.1564	-0.1441	True
KNN	NB	-0.0839	0.001	-0.0901	-0.0778	True

With each model comparison on a confidence interval of 0.05, each model was found to have scores which differed significantly from one another. Meaning that each model can be confidently evaluated as separate and the results should be viewed as independent from one another.

Discussion and Conclusion

A board game database of publicly amassed and available data was analyzed by three models: Decision Tree, K nearest neighbors, and Naïve Bayes, to determine if the available data could confidently predict highly rated boardgames and subsequently determine the features of greatest importance. Due to a lack of prior exploratory data in this specific area, much of the results are novel.

Each of the models performed quite differently from one another, with the Decision tree classifier performing the best of the three (0.75 versus 0.69 for K-NN and 0.61 for Naïve Bayes). Interestingly, even though there were significant differences in the results, each classification model still struggled to identify and predict accurately between high and medium rated board games. This may mean that there is not a strong enough difference in the dataset that was used to be able to make that determination. Both the Decision tree and Naïve Bayes classifiers were very good at predicting medium rated board games, however, achieving a result of 0.69 precision in each case. The low results for knn match the expectation of the known limitations of the model, where if there is a noisier dataset, knn performance will erode.

Each model also had a difference in performance by the length of time that they took to be executed. The Naïve Bayes model completed near instantaneously upon execution and was by far the most expedient. Both Knn and Decision tree models took longer to execute (roughly 30 seconds to almost 2 minutes for each). However, Knn suffered from the inability to be properly tuned due to the extreme length of time it took to execute the code. This is unsurprising as it is well known that as k gets larger in knn, the longer the code will take to run.

Given this information and limitations, it would appear that the decision tree model is the ideal candidate to conduct predictions of the dataset, under these circumstances. The model appears to best determine whether a board game will be rated low (<5.0) or anything above that (≥ 5.1) as the model tends to degrade in quality

when looking at the highest class. The highest predictor of the medium class appeared to be the stats.useraverage ≤ 0.5 , then stats.bayesaverage and the year published as the highest influencers. Surprisingly, play length or number of players and optimal playing groups had much lower influences and were otherwise unobserved in this study.

This area of research is scant, however it would still benefit from the type of exploration employed Khan et al. (2020) in exploring genres of board games by tokenization and use of Naïve bayes after grouping the genres and expanding on genre themes and game descriptions to capture other genres not listed. Additionally, forward selection or backward elimination would be suited to be used to better determine the influence of the qualities of the board games.

References

1. Danescu-Niculescu-Mizil C., Kossinets G., Kleinberg J., & Lee L. (2009). How opinions are received by online communities: a case study on amazon.com helpfulness votes. *WWW '09: Proceedings of the 18th international conference on World wide web*, 141–150, New York, NY, USA, ACM.
2. Ghose A., & Ipeirotis P. G. (2007). Designing novel review ranking systems: predicting the usefulness and impact of reviews. *ICEC '07: Proceedings of the ninth international conference on Electronic commerce*, 303-310.
<https://dl.acm.org/doi/abs/10.1145/1282100.1282158>
3. Guo Y., Barnes S. J., & Jia Q. (2017). Mining meaning from online ratings and reviews: Tourist satisfaction analysis using latent dirichlet allocation. *ScienceDirect*, 59, 467-483.
<https://www-sciencedirect-com.ezproxy.lib.rverson.ca/science/article/pii/S0261517716301698>
4. Jae-Won K., Wi J., Jang S., & Kim Y. (2020). Sequential Recommendations on Board-Game Platforms. *Symmetry*, 12(2), 210-225. <https://www.mdpi.com/2073-8994/12/2/210>
5. Kamal A., Saaidin S., & Kassim M. (2020) Recommender System: Rating predictions of Steam Games Based on Genre and Topic Modelling. *2020 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, 212-218.
<https://ieeexplore.ieee.org/abstract/document/9140194>
6. Khan E. M., Mukta S. H., Ali E. M., & Mahmud J. (2020). Predicting Users' Movie Preference and Rating Behavior from Personality and Values. *ACM Transactions on Interactive Intelligent Systems* 10(3), 1-25. <https://dl.acm.org/doi/abs/10.1145/3338244>
7. Roeder, O. (2015, August 18). Crowdfunding Is Driving A \$196 Million Board Game Renaissance. *FiveThirtyEight*. <https://fivethirtyeight.com/features/crowdfunding-is-driving-a-196-million-board-game-renaissance>

8. Siersdorfer S., Chelaru S., Nejdl W., & Pedro J. S. (2010)How useful are your comments?: analyzing and predicting youtube comments and comment ratings. *WWW '10: Proceedings of the 19th international conference on World wide web*. 891-900.
<https://dl.acm.org/doi/abs/10.1145/1772690.1772781>
9. Steck H. (2013). Evaluation of recommendations: rating-prediction and ranking. *RecSys '13: Proceedings of the 7th ACM conference on Recommender systems*, 213-220.
<https://dl.acm.org/doi/abs/10.1145/2507157.2507160>
10. Turney P. D. (2002). Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 417-424.
<https://arxiv.org/abs/cs/0212032>