4. Modeling

4.1 Pre-Processing

The original dataset is split into two different datasets, one for the content-based filtering implementation and one for the collaborative-based filtering implementation.

For the content-based filtering dataset, the original dataset is reduced to include only one record for each unique game (duplicate game records are filtered out), for a total of 4223 records. Each unique game has its own similar list of similar games, which is used to evaluate the content based filtering recommendations. All of the game's metadata is combined into one feature, the bag of words.

For the collaborative-based filtering dataset, in order to ensure similarities between users, the original dataset is reduced to include only games that have been reviewed at least 2 times, and users that have made at least 2 reviews. This reduced the dataset from 24023 reviews to 15809 reviews, which is now ready to be utilized by the matrix factorization and clustering models to predict user ratings for unrated games. Because matrix factorization assigns a value of 0 to missing values in the user-item rating matrix, each record's score is increased by a value of 1, so that the ratings range from 1.0 to 6.0, to prevent reviews with a score of 0 to change to a predicted value.

4.1 Content-Based Filtering

The approach that the content-based filtering will recommend games based on the initial game's similarity to other games, will rely on the information from the game's metadata (genre, theme, description. etc.). Here, the TF-IDF algorithm will be implemented to weigh words in each game's bag of words, which is an amalgamation of all of the game's metadata data into a single feature, and assign importance to those words based on the number of times they appear in the bag of words. Each game is stored as a vector of the game's attributes, where the angle between each vector, in an n-dimensional space, is calculated to determine the similarity between each game's vector. The measure of similarity that is used is called cosine similarity, because when the angle between two vectors decreases, the value of cosine will increase, signifying more similarity between the vectors, or games.

Because the 'similar games' feature will evaluate how accurate the TF-IDF method recommendations, we want to determine which combination of features will create the most accurate recommender system. The metrics measured will be precision, recall, and F1 score, in which each game's precision, recall, and f1 score is summed and averaged over the total number of unique games. Ideally we don't want to recommend too few or too many games for the user to choose from, so the baseline number of recommendations will be 10, for which:

1. Precision = # of games from the recommendations that are in the similar games list / 10
2. Recall = # of games from the recommendations that are in the similar games list / # of games in the similar games list
3. F1 = $2 * (\frac{precision * recall}{precision + recall})$

We start with all features combined to make the bag of words, calculate the metrics and accuracy of the recommendations, then systematically remove one feature from the bag of words. If the removed feature creates more accurate recommendations, it stays removed, otherwise, it remains in the bag of words. The order of which feature is removed was determined using the previous analysis of the composition of the similar games list (). The results of the TF-IDF recommendations are shown below:

Table. Results of TF-IDF recommendations for each bag of words combination

Bag of Words includes:	Avg. Precision	Avg. Recall	Avg. F1
Every feature	0.087	0.191	0.103
Every feature except characters	0.088	0.193	0.104
Every feature except characters and game body*	0.109	0.206	0.122
Every feature except characters, game body, franchises	0.110	0.209	0.123
Every feature except characters, game body, franchises, publishers	0.111	0.211	0.125

* "Game body" refers to full length description of the game

The features used (combined for the bag of words), for most accurate recommendations, were genres, themes, concepts, developers, platforms, game_deck (short summary of game). Any feature, or features, removed from this bag of words will lower the accuracy metrics, so we stop the feature selection here. The best precision, 0.111, indicates that the system, on average, can recommend at least 1 game in each game’s similar games list, and the best recall, 0.211, indicates that the system, on average, can recommend at least 20% of the games in each game’s similar games list. Again, this does not mean that the recommendations system is not accurate, with the inclusion of all features, as accuracy, for the sake of this dataset, is judged by the games within the similar games feature list.

To display an example of what the system recommends for the game Pac Man:

Table. Example recommendations for Pac Man

Similar Games list for Pac-Man
Lock-n-Chase
Lady Bug
Katamari Damacy
Spore
Mario Bros.
Dig Dug
Bit Boy!!
Bit Trip Void
The Legend of Zelda: Spirit Tracks
Jungler
Gubble
Monaco: What’s Yours Is Mine
Grand Theft Auto
Wolfenstein-3D
Metal Gear
Galaxian

Space Invaders

Recommended Games	In Similar Games list?
Dig Dug	Yes
Ms. Pac Man	No
Galaxian	Yes
Galaga	No
Centipede	No
Mario Bros.	Yes
Frogger	No
Pac Man Championship Edition	No
Tetris	No
Space Invaders	Yes

4.2 Collaborative-Based Filtering

Collaborative-based filtering involves the prediction of unknown ratings through the similarities between users. Here, model-based algorithms, such as clustering (KNN) and matrix factorization (SVD, NMF) are implemented with the package: Surprise. 11 algorithms are run, using cross validation, on the dataset to determine which algorithms are most accurate with predicting ratings, using RMSE and MAE as metrics. The results are shown below:

Algorithm	test_rmse	test_mae	fit_time	test_time
SVDpp	0.876658	0.664882	7.524015	0.250665
SVD	0.885243	0.671107	1.715744	0.075928
BaselineOnly	0.891370	0.673919	0.101063	0.024599
KNNBaseline	0.935266	0.693460	0.165082	0.315176
CoClustering	0.984642	0.714413	1.319195	0.056860
KNNBasic	0.995987	0.712726	0.131267	0.310606
KNNWithMeans	0.998343	0.729940	0.228719	0.340424
KNNWithZScore	1.022443	0.745451	0.382309	0.289892
SlopeOne	1.075540	0.790935	0.152590	0.161901
NMF	1.149029	0.905406	1.807499	0.040558
NormalPredictor	1.321550	1.029843	0.019945	0.043218

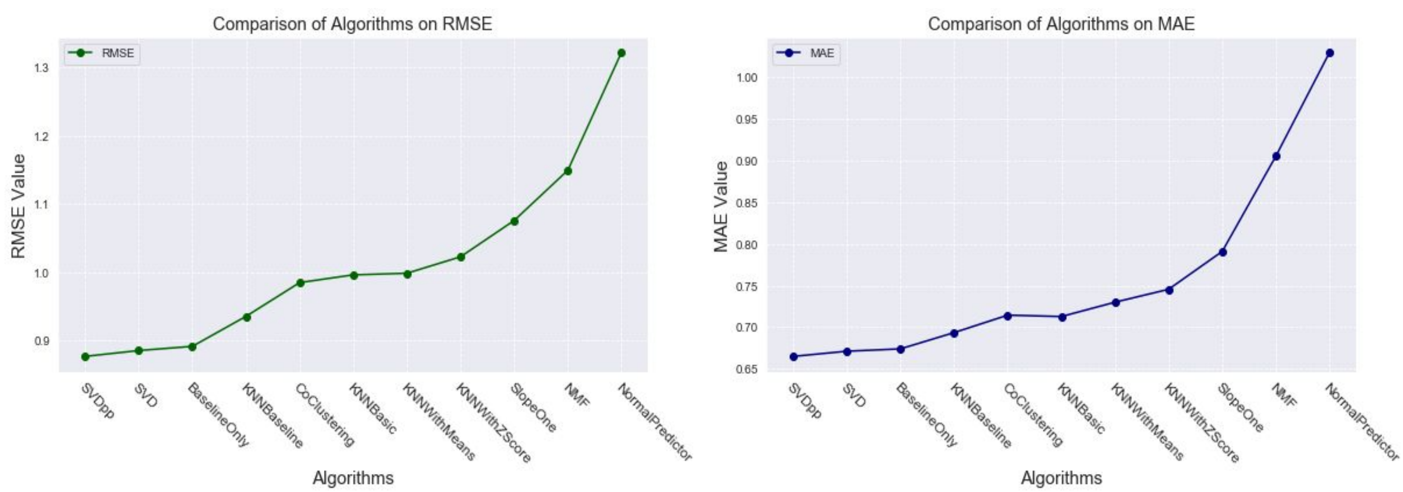

Figure. Results and visualization of each algorithm's cross validation on dataset

The algorithm with the lowest RMSE and MAE, or with most accurate predictions, is the matrix factorization algorithm SVDpp, a derivative to the second best algorithm, SVD. SVDpp achieves better predictive accuracy than normal SVD, due to the addition of implicit feedback information. This means that a user exhibits bias or preference to games that they have reviewed, and these preferences are taken into consideration within the algorithm. SVDpp and SVD will be used to train the dataset and predict ratings for a test set to see how the algorithm can handle unseen data.

Next, GridSearch cross validation is implemented to determine the best parameters for both SVDpp and SVD, which were determined as follows:

1. n_epochs: 25
2. lr_all: 0.01
3. reg_all: 0.4

With this information, the dataset is split into a training and testset, and both SVDpp and SVD, with best parameters, are used to fit the training set, and used to predict for the testset. The results are shown below:

Table. SVDpp and SVD evaluation of test data

Algorithm	RMSE	MAE
SVDpp	0.8682	0.6662
SVD	0.8708	0.6679

Looking at the residual plot:

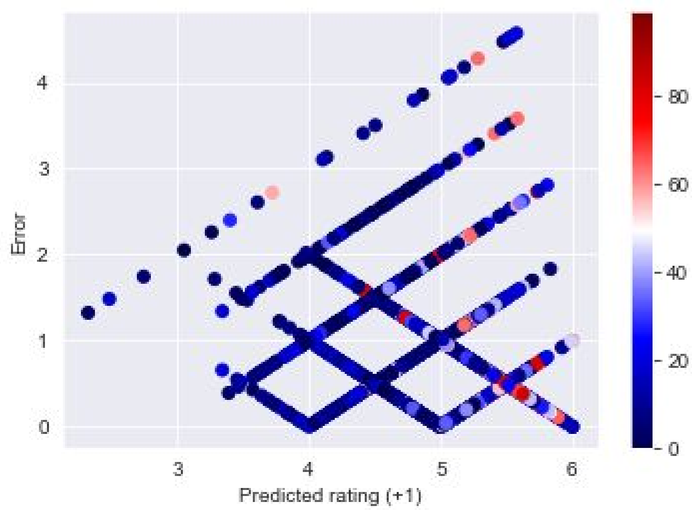

Figure. Residual plot for SVDpp test set

The system overpredicts, especially for 0 and 1 ratings, which can be attributed to the highly positive skewed distribution of ratings. The residuals would be a lot more balanced, across an error of 0, if more negatively rated games were included in the dataset, and help alleviate the overprediction problem, by balancing the distribution of ratings. Because of this issue, the content based filtering method in recommending games is preferable to use for this dataset, as it does not rely on other users to make recommendations.

5. Further Research and Recommendations

5.1 Future Work

For the content-based filtering method,while the game features provide substantial information in describing each of the 4223 unique games, the recommendations can always expand with added new games and their game information. A new evaluator list for the recommendations should also be determined, possibly through surveying gamers to determine what they feel are the most similar games to the games they have played, and strengthening how accurate the system will recommend games. Multiple surveys of similar games should provide enough evidence to determine a consensus list of similar games that can be used to evaluate the system.

For the collaborative-based filtering method, the dataset lacks negatively rated games, causing the system to overpredict ratings, so adding more reviews for these games may help alleviate this issue, but if not, will increase the amount of reviews for each game to make better rating predictions. Another idea that could be implemented is a hybrid system that combines collaborative based filtering with content based filtering, and add sentiment analysis, which would use the user review feature to recommend games tailored towards the individual querying the system.

5.2 Recommendation to Clients

Based on the findings from the data exploration and modeling, it is recommended that:

1. Gamers should use the content based filtering method of recommending games over the collaborative based method, as the dataset is highly skewed towards positive ratings, and games rated negatively will be overpredicted upon.
2. If using the content based filtering method, gamers should be aware that features considered are genres, themes, concepts, platforms, developers, and short game summary, as it produced the best metrics when evaluated by the similar games list. If the gamer enjoys their game due to other features, such as their favorite character, or franchise, the system is unlikely to recommend a sequel/prequel/spin-off to their game.
3. Even though a few features are considered by the content-based filtering system, this does not mean that the recommendations are correct, as the features were determined through evaluating the system with a man-made list of similar-games to the target game. The gamer should feel that these are the most similar games to

the target game, as different lists of games should be used to evaluate the system, in terms of how well the system can recommend similar games.

6. Conclusion

6.1 Data Exploration Conclusion

To summarize the findings from the exploratory data analysis of the review/game data:

1. **Ratings and Reviews:** The dataset is heavily skewed towards positively rated games (many 4's and 5's). Many users have only rated one game, and many games have only been reviewed upon one time.
2. **Genres, Themes, and Concept:** Majority of games reviewed upon were of genres: Action and Action-Adventure, themes: Fantasy and Sci-Fi, concepts: Achievements and Polygonal-3D. Highly associated popular genres include: Shooter with Action, popular themes: Post-Apocalyptic with Sci-Fi. The percentage of games, in the similar games list, that are of the same genre, theme, or concept to the target game is relatively high, compared to the others features, indicating these features will be highly considered when recommending games evaluated by the similar games list. This does not mean that the other features, in general, should not be used to recommend games.
3. **Characters and Franchises:** The percentage of games, in the similar games list, that are of the same character or franchise is extremely low, indicating that these features should not be included in recommending game.by the content-based system. Once again, this does not mean these features, in general, should not be included, but for the purposes of evaluating the system with the similar-games list, they may not be.
4. **Platforms, Publishers, and Developers:** Strong association between most consoles and PC exists, as most games are available digitally on PC, and games overlap between Playstation and Xbox (few exclusives). The percentage of games, in the similar games list, that are of the same platform as the target game is relatively high, indicating that it will be an important feature to include when recommending games. However, the percentage of games, in the similar games list, that are of the same publisher, or developer, is extremely low.
5. **Term Frequency:** Looking at the frequency of terms, the popular terms all pertain to general gaming language, such as: 'game', 'player', 'character', 'xbox 360', etc. Action terms dominate: 'weapon', 'enemies', 'first person shooter', 'real time combat', etc.

6.2 Modeling Conclusion

To summarize the findings from content based filtering and collaborative based filtering modeling:

1. **Content based filtering:** The optimal combination (bag of words) of features, when evaluated by the similar games list, were genres, themes, concepts, platforms, developer, and short game summary. The accuracy metrics for this combination of features, to recommend games, were a precision of 0.111 (average of 1 game in similar games list, out of 10 recommendations), a recall of 0.211 (average of 20% of the games in similar games list are recommended), and the harmonic mean of the precision recall, the F1 score, of 0.125.
2. **Collaborative based filtering:** Out of all CF algorithms ran on dataset, SVDpp and SVD produced the lowest error, RMSE and MAE, indicating they are the optimal algorithms to run for this dataset. GridSearchCV determined the optimal parameters for SVDpp and SVD, and the model was fit on a training set, and evaluated using the test set, for which the errors were: SVDpp RMSE and MAE of 0.8682 and 0.6662 respectively, SVD RMSE and MAE 0.8708 and 0.6679. According to the residual plot, the system overpredicts as a result of the uneven distribution of the ratings (heavily skewed to positive ratings). In this case, content-based filtering is the preferable method to recommending the games.

1. Executive Summary

- Using game review data, I constructed a recommendation system, using content-based filtering and collaborative based filtering methods. 24023 game reviews and ratings were extracted from the video game database GiantBomb, along with the reviewed game's features, such as genre, themes, concepts, description, etc. To evaluate the content-based filtering method, an additional feature was extracted, similar games, which displays a list of games that the GiantBomb staff deemed similar to its respective game. There were a total of 6561 users reviewing 4223 games, spanning from July 2008 to September 2019.
- A relatively high percentage of games, in the similar games list, were of the same genre, theme, concept, platform, whereas an extremely low percentage of games were of the same character, franchise, publisher, and developer, when looking at the most popular values for these features.
- For content-based filtering, the best combination of features, when evaluated by the similar games list, was determined to be genres, themes, concepts, platforms, and short game summary, yielding a precision of 0.111 (average of 1 game in similar games list, out of 10 recommendations), a recall of 0.211 (average of 20% of the games in similar

games list are recommended), and the harmonic mean of the precision recall, the F1 score, of 0.125.

- For collaborative-based filtering, the best algorithm that evaluating the overall dataset was determined to be the matrix factorization algorithms, SVDpp and SVD. Both algorithms fitted the training data, and evaluated using the test data, for which the errors were: SVDpp RMSE and MAE of 0.8682 and 0.6662 respectively, SVD RMSE and MAE 0.8708 and 0.6679. However, the system overpredicts ratings, as a result of the uneven distribution of the ratings (heavily skewed to positive ratings). In this case, content-based filtering is the preferable method to recommending the games.