Analyzing Genre-specific Lexical and Thematic Patterns in Video Game Reviews

A Machine Learning Approach Using TF-IDF Vectors and Steam User-generated Tags

Nico Benz

3583917

nico.benz@studserv.uni-leipzig.de

Project report for the module

10-207-0101

Current Trends in Digital Humanities: Computational Game Studies

/nicobenz/GameStudies-SteamPredictions

Computational Humanities
Institute of Computer Science
Leipzig University

September 8, 2023

Abstract

This paper focuses on Computational Game Studies as an increasingly popular branch of Digital Humanities. Review texts of video games along with user generated genre tags for each game have been extracted from the video game platform Steam resulting in an exhaustive corpus containing more than one billion token, providing a comprehensive foundation for analysis. Machine learning models including Naive Bayes, Logistic Regression, Support Vector Machine and Random Forest have been trained on review texts along with the genre tags associated with the game that each particular review is belonging to. The findings show that these models are capable of predicting a games genre given review texts, highlighting the potential presence of underlying genre-specific lexical features embedded within these texts. Analysis of TF-IDF vector scores also show high ranking of tokens related to topics or game content within each genre, further supporting the hypothesis of genre specific lexical features in video game reviews.

Keywords: Computational Game Studies, Natural Language Processing (NLP), Machine Learning (ML), Computational Linguistics (CL), Steam Reviews, Label Prediction, Classifier Model

Contents

List of Tables

List of Figures

1 Introduction

2 Related work

As already mentioned, label classification tasks are thoroughly researched up to this date, including research questions similar to the one of this paper.

3 Methodology

In the past countless researchers could show that machine learning models are very capable of performing label classification in various situations and for various types of texts.

4 Data overview

On the Steam platform users have the ability to write a public text review for any game. These reviews can be retrieved using the official Steam API. Given the vast amount of review data available on steam, this process took some time with a python script generating API requests and saving the retrieved reviews non-stop for more than two months between May and July 2023. The user generated genre tags were not part of the data retrieved from the API and have been retrieved using a custom script crawling each Steam games web page and scraping the tags from the HTML content. Both processes were running in parallel but the crawling of the tags finished much faster, because of the very low amount of target data. This resulted in very small discrepancies between the review and the tag dataset because of games being removed from or added to Steam in the time window when the tag crawling was already finished but the review extraction not yet. Affected games where either the review set or the tag set was missing were not used in the combined corpus.

4.1 Games

In total data of 134,076 games was downloaded which includes all games present on the platform at that time. Of these games all of their reviews have been downloaded. In some rare cases retrieval of all reviews of a certain game was not possible due to the API not giving a new cursor to fetch the next batch of reviews. After 10 tries of not getting a new cursor the script moved on to the next game, saving only the reviews gathered up to that point. This problem only occurred for games with several million reviews and only after at least one million reviews have been downloaded of that game. Because such great amounts of reviews for a single game would not be used for training to avoid strong bias or imbalance of the dataset anyway, this will not pose a challenge. More on sampling in a later section. See figure 1 for an overview of the distribution of games and review numbers.

4.2 Reviews

Among all these games the numbers of reviews for each game are not evenly distributed, as was expected. The biggest portion of games do not have reviews at all, while some games have more than one million reviews. These reviews also vary strongly in their length but all withing the dimensions set by Steam which needs a review to be between 5 and 8,000 characters. The Steam API returns reviews as batches in a JSON format containing the reviews along with some anonymous meta data about the author and the review. For some basic metrics of the reviews see table 1.

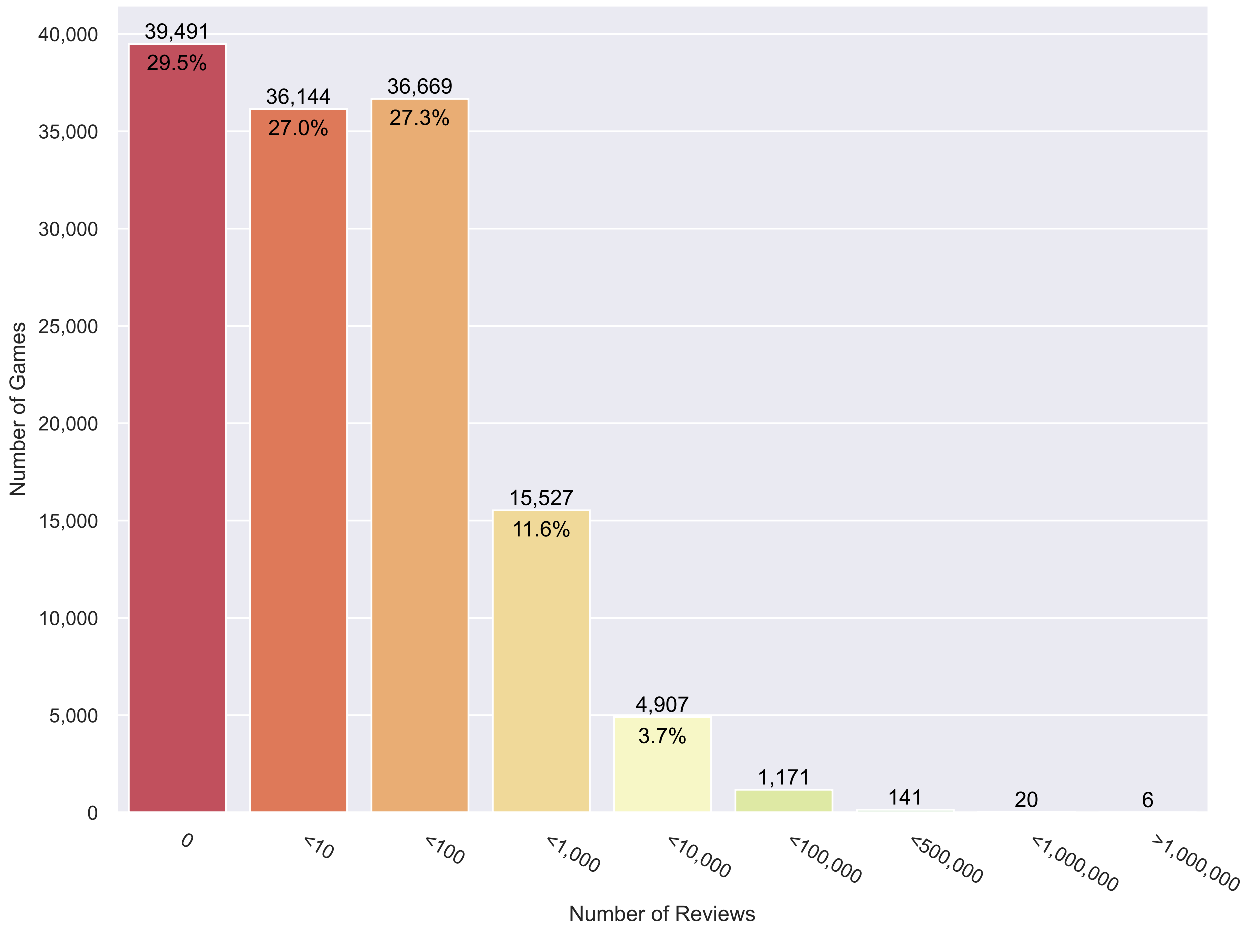

Figure 1: Distribution of Reviews Across Games

	Total	Only English
Reviews	100,855,789	46,616,033
Tokens	15,023,225,453	10,643,950,482
Mean Review Size	148.96	228.33
Median Review Size	20	59

Table 1: Review Metrics

4.3 Genres

The term genre is used very loosely in this paper. On Steam, users have the ability to give tags or key words to games or to upvote tags that are already given to that game, incrementing the tags counter. Through this mechanic, games have a list of user given tags along with the number of users that have given or upvoted a tag. If a game has more than 20 tags associated with it, only the 20 most common tags will be displayed on Steam. While most tags correspond to traditional genre labels (e.g. adventure, strategy, RPG) some of them focus more on features that could be present in any genre (e.g. early access, free to play, VR). There are also tags where it is not entirely clear whether they correspond to a genre or not (e.g. indie, exploration, casual) because of the fluidity of emerging new genres and subgenres. To avoid biased selection this paper will call all user generated tags genres. Further sampling and clustering of these tags into more traditional genres is always possible for future research at a later date. See figure 2 for an overview of genres that are found within the 5 most common genres of all games in the corpus.

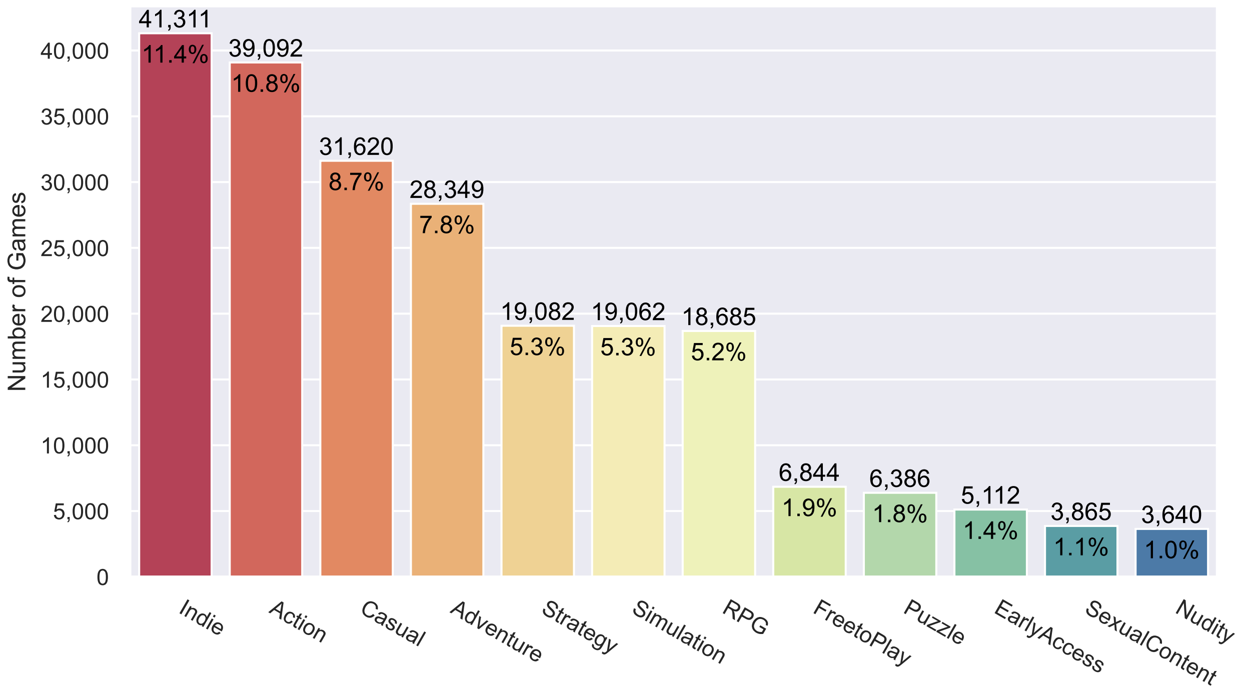

Figure 2: Number of Games for the Ten Most Common Genres

5 Experimental design

5.1 Corpus generation

The review data was gathered utilising Python scripts[1] along with the Steam API. The genre data was not part of the Steam API and was therefore crawled through a custom script accessing each games page on the Steam website and downloading its HTML content for scraping in a later step. Review and genre data has been combined by saving it in a custom database storing information like game id, review id, review text and most common genre tags for every review for faster speeds when creating data samples. These samples were created with multiple restrictions to attain as much balance as possible while using as much data as possible. Only games were included that had at least 10 reviews and each review was randomly chosen and had to be between 20 and 1000 token while collecting maximum of 1000 reviews per game. Collected reviews where tokenised, cleaned from stop words and custom stop words, special characters and any other noise like ASCII art, and then lemmatised. Custom stop words were collected through comparison of most prominent token in earlier corpora using TF-IDF scores among all genres present in the corpora. The 50 most common tokens for all genres have been collected and counted. Among these token, only those present in at least three genres have been removed. Those consisted mostly of tokens belonging to review as a text genre like *recommend*, *feel* or *like* or of very generic adjectives like *good*, *bad* or *fun*. Tokens belonging to the broad field of video games like *game* or *play* were also added to the custom stop words. Content-specific words like *level* or *gameplay*, however, have not been removed. For a full list of all custom stop words that were removed see appendix A. TF-IDF vectors were created off the cleaned reviews and then corrected for imbalances using Synthetic Minority Oversampling Technique (SMOTE), since although the same amount of review texts has been selected, the average text length differed slightly in each genre label. SMOTE was able to correct the imbalances by raising the minority classes to the same level as the majority classes by introducing slight amounts of synthetic data. The results were then passed on to training.

5.2 Model training

The already prepared dataset was separated into TF-IDF vectors and genre labels, and a 80/20 split has been performed for separating the training from the test set. Multiple classifier models were trained, including Naive Bayes, Logistic Regression, Random Forest and Support Vector Machine. After training, relevant metrics like accuracy, recall, precision and F1 score were collected.

6 Results and discussion

	Mean	Naive Bayes	Logistic Regression	Random Forest
Recall	0.66	0.68	0.68	0.61
Precision	0.65	0.67	0.68	0.61
F1 Score	0.65	0.67	0.68	0.61
Support	50000.0	50000.0	50000.0	50000.0

Table 2: Classifier Model Metrics Across All Genres

[1] See the repository linked on the front page for access to all scripts and code used.

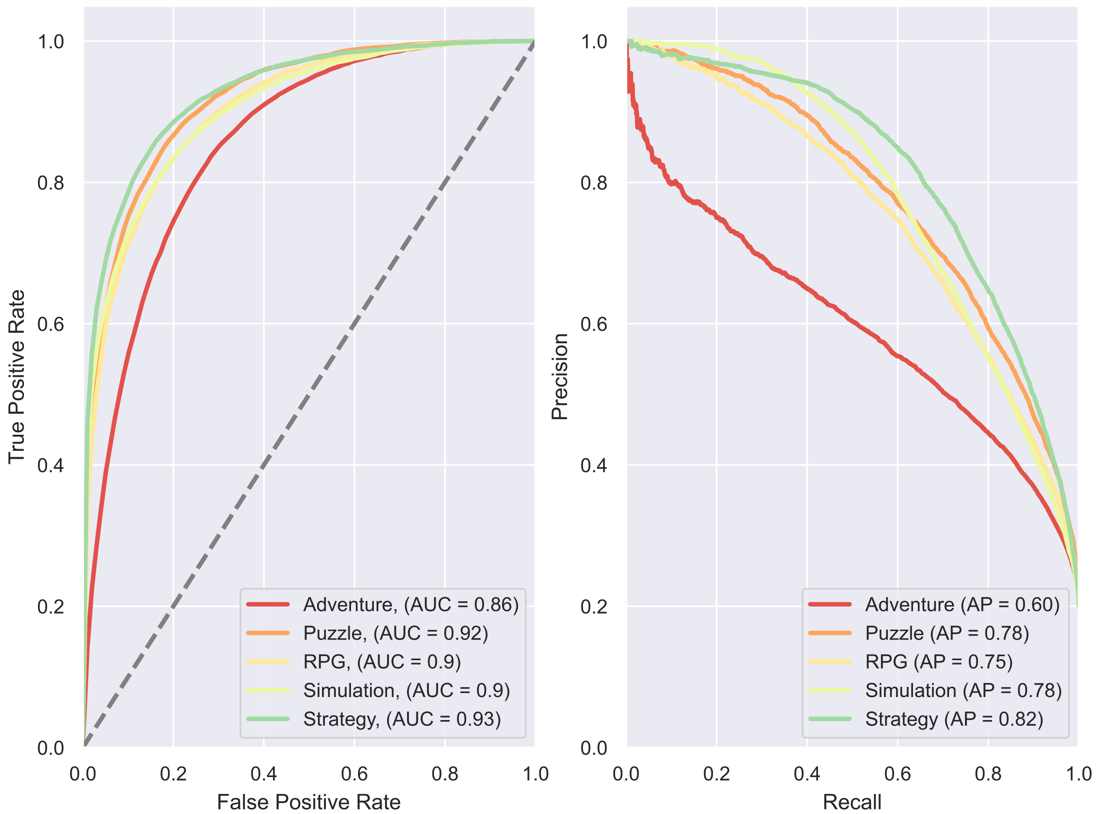

Figure 3: ROC Curve OvR for Naive Bayes

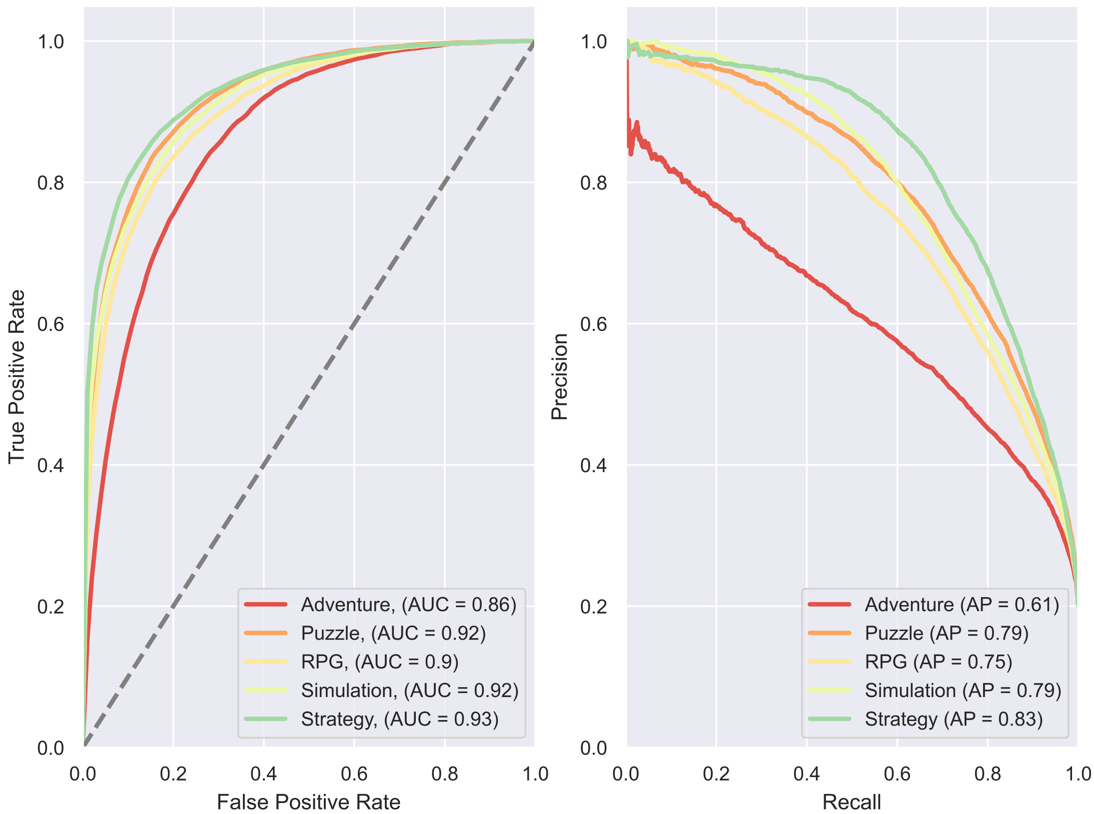

Figure 4: ROC Curve OvR for Logistic Regression

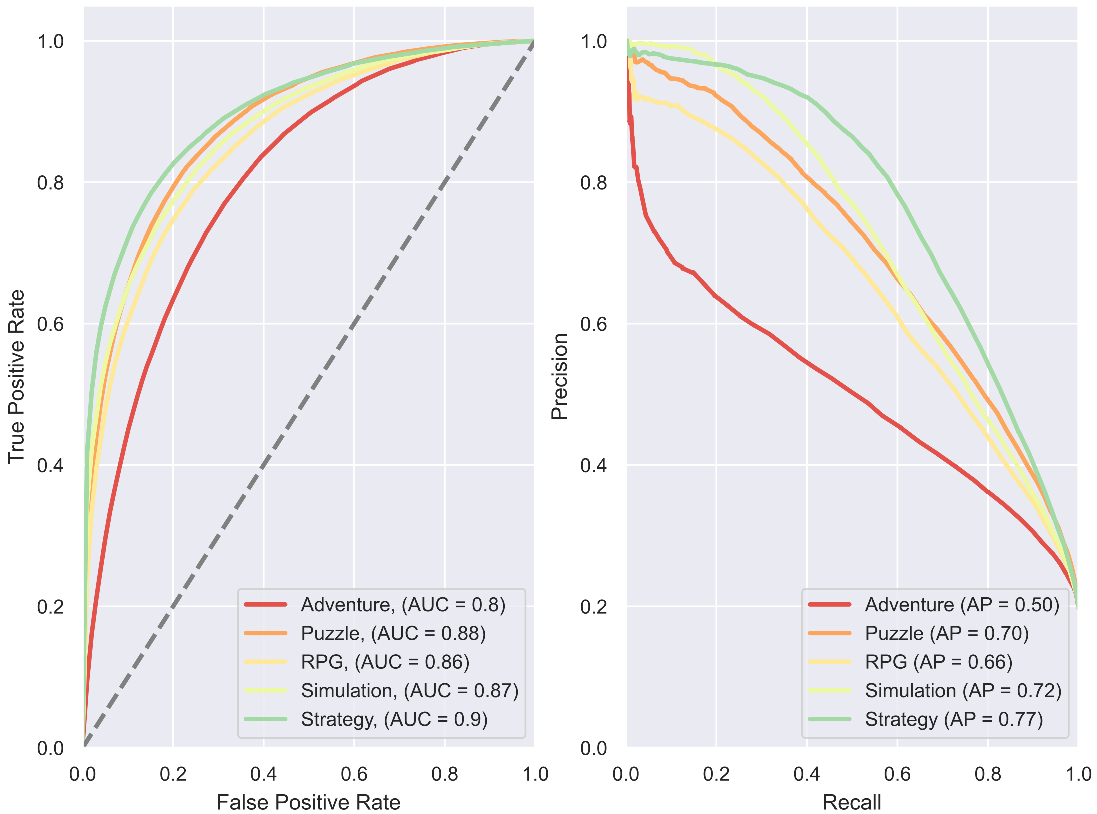

Figure 5: ROC Curve OvR for Random Forest

	Average	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Naive Bayes						
Recall	0.68	0.55	0.7	0.68	0.74	0.74
Precision	0.67	0.62	0.71	0.66	0.65	0.75
F1 Score	0.67	0.58	0.71	0.67	0.69	0.74
Support	50000.0	9912.0	10200.0	9914.0	10036.0	9938.0
Logistic Regression						
Recall	0.68	0.57	0.71	0.69	0.7	0.75
Precision	0.68	0.59	0.71	0.67	0.71	0.73
F1 Score	0.68	0.58	0.71	0.68	0.71	0.74
Support	50000.0	9912.0	10200.0	9914.0	10036.0	9938.0
Random Forest						
Recall	0.61	0.51	0.62	0.63	0.63	0.69
Precision	0.61	0.5	0.66	0.59	0.64	0.69
F1 Score	0.61	0.5	0.64	0.61	0.64	0.69
Support	50000.0	9912.0	10200.0	9914.0	10036.0	9938.0

Table 3: Classifier Model Metrics Across All Folds

	Average	Adventure	Strategy	Simulation	RPG	Puzzle
Naive Bayes						
Recall	0.68	0.55	0.69	0.68	0.73	0.73
Precision	0.67	0.6	0.71	0.66	0.65	0.75
F1 Score	0.67	0.58	0.7	0.67	0.69	0.74
Support	50000.0	10000.0	10000.0	10000.0	10000.0	10000.0
Logistic Regression						
Recall	0.68	0.57	0.71	0.68	0.7	0.75
Precision	0.68	0.59	0.71	0.67	0.71	0.73
F1 Score	0.68	0.58	0.71	0.68	0.7	0.74
Support	50000.0	10000.0	10000.0	10000.0	10000.0	10000.0
Random Forest						
Recall	0.61	0.5	0.61	0.63	0.63	0.69
Precision	0.61	0.49	0.65	0.59	0.64	0.69
F1 Score	0.61	0.5	0.63	0.61	0.64	0.69
Support	50000.0	10000.0	10000.0	10000.0	10000.0	10000.0

Table 4: Classifier Model Metrics Across All Genres

7 Conclusion

Adventure	Strategy	Simulation	RPG	Puzzle	
story, 0.25	units, 0.15	route, 0.15	story, 0.2	levels, 0.22	
short, 0.13	strategy, 0.14	vr, 0.15	combat, 0.16	level, 0.17	
music, 0.11	tower, 0.12	work, 0.12	characters, 0.13	story, 0.16	
gameplay, 0.11	ai, 0.11	better, 0.11	character, 0.12	music, 0.14	
characters, 0.11	gameplay, 0.11	buy, 0.11	hours, 0.11	simple, 0.12	
graphics, 0.1	cards, 0.1	real, 0.1	gameplay, 0.11	easy, 0.12	
level, 0.1	campaign, 0.1	things, 0.09	system, 0.1	gameplay, 0.11	
experience, 0.1	defense, 0.1	money, 0.09	better, 0.1	short, 0.11	
art, 0.1	turn, 0.1	price, 0.09	far, 0.09	hours, 0.1	
point, 0.09	hours, 0.1	experience, 0.09	world, 0.09	hard, 0.09	
interesting, 0.09	card, 0.1	free, 0.08	buy, 0.09	art, 0.09	
controls, 0.09	better, 0.09	graphics, 0.08	interesting, 0.09	achievements, 0.09	
price, 0.09	buy, 0.09	train, 0.08	content, 0.08	price, 0.09	
better, 0.08	player, 0.09	add, 0.08	enemies, 0.08	mechanics, 0.09	
character, 0.08	war, 0.08	sim, 0.08	things, 0.08	challenging, 0.09	
⋮	⋮	⋮	⋮	⋮	⋮

Table 5: 20 Most Prominent Tokens with TF-IDF Scores by Genre

Token	Adventure	Strategy	Simulation	RPG	Puzzle
story	0.25	-	0.06	0.2	0.16
gameplay	0.11	0.11	0.06	0.11	0.11
hours	0.08	0.1	0.07	0.11	0.1
better	0.08	0.09	0.11	0.1	0.07
level	0.1	0.08	-	0.08	0.17
buy	0.07	0.09	0.11	0.09	0.06
graphics	0.1	0.08	0.08	0.07	0.08
price	0.09	0.07	0.09	0.07	0.09
work	0.07	0.08	0.12	0.07	0.06
things	0.08	0.07	0.09	0.08	0.07

Table 6: 10 Most Prominent Tokens with TF-IDF Scores across Genres

References

Aarseth, E. (1997). *Cybertext: Perspectives on ergodic literature*. JHU Press.

Aarseth, E. (2001). Computer game studies, year one. *Game studies*, *1*(1), 1–15.

Agarwal, A., Das, R. R., & Das, A. (2021). Machine learning techniques for automated movie genre classification tool. *2021 4th International Conference on Recent Developments in Control, Automation & Power Engineering (RDCAPE)*, 189–194. https://doi.org/10.1109/RDCAPE52977.2021.9633422

Apperley, T. H. (2006). Genre and game studies: Toward a critical approach to video game genres. *Simulation & Gaming*, *37*(1), 6–23. https://doi.org/10.1177/1046878105282278

Atdag, S., & Labatut, V. (2013). A comparison of named entity recognition tools applied to biographical texts. *2nd International Conference on Systems and Computer Science*. https://doi.org/10.1109/icconscs.2013.6632052

Atkins, B. (2003). *More than a game*. Manchester University Press. https://doi.org/10.7228/manchester/9780719063640.001.0001

Bahuleyan, H. (2018). Music genre classification using machine learning techniques. https://doi.org/10.48550/ARXIV.1804.01149

Cui, L., Wu, Y., Liu, J., Yang, S., & Zhang, Y. (2021). Template-based named entity recognition using bart. https://doi.org/10.48550/ARXIV.2106.01760

Finn, A., & Kushmerick, N. (2006). Learning to classify documents according to genre. *Journal of the American Society for Information Science and Technology*, *57*(11), 1506–1518. https://doi.org/10.1002/asi.20427

Göring, S., Steger, R., Rao Ramachandra Rao, R., & Raake, A. (2020). Automated genre classification for gaming videos. *2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP)*, 1–6. https://doi.org/10.1109/MMSP48831.2020.9287122

Goyal, A., Gupta, V., & Kumar, M. (2018). Recent named entity recognition and classification techniques: A systematic review. *Computer Science Review*, *29*, 21–43. https://doi.org/10.1016/j.cosrev.2018.06.001

Gupta, S., Agarwal, M., & Jain, S. (2019). Automated genre classification of books using machine learning and natural language processing. *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*. https://doi.org/10.1109/confluence.2019.8776935

Heintz, S., & Law, E. L.-C. (2015). The game genre map: A revised game classification. *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play*, 175–184. https://doi.org/10.1145/2793107.2793123

Jiang, Y., & Zheng, L. (2023). Deep learning for video game genre classification. *Multimedia Tools and Applications*, *82*(14), 21085–21099. https://doi.org/10.1007/s11042-023-14560-5

Katharina Sienčnik, S. (2015). Adapting word2vec to named entity recognition. *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*, 239–243.

Kumar, S., Kumar, N., Dev, A., & Naorem, S. (2022). Movie genre classification using binary relevance, label powerset, and machine learning classifiers. *Multimedia Tools and Applications*, *82*(1), 945–968. https://doi.org/10.1007/s11042-022-13211-5

Mansouri, A., Affendey, L. S., & Mamat, A. (2008). Named entity recognition approaches. *International Journal of Computer Science and Network Security*, *8*(2), 339–344.

Tamla, P., Freund, F., & Hemmje, M. (2020). Supporting named entity recognition and document classification in a knowledge management system for applied gaming. *Proceedings of the 12th*

International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management. https://doi.org/10.5220/0010145001080121

Viggiato, M., Lin, D., Hindle, A., & Bezemer, C.-P. (2022). What causes wrong sentiment classifications of game reviews? *IEEE Transactions on Games*, *14*(3), 350–363. https://doi.org/10.1109/TG.2021.3072545

Windleharth, T. W., Jett, J., Schmalz, M., & Lee, J. H. (2016). Full steam ahead: A conceptual analysis of user-supplied tags on steam. *Cataloging & Classification Quarterly*, *54*(7), 418–441. https://doi.org/10.1080/01639374.2016.1190951

Zagal, J. P., Tomuro, N., & Shepitsen, A. (2011). Natural language processing in game studies research. *Simulation & Gaming*, *43*(3), 356–373. https://doi.org/10.1177/1046878111422560

Appendix A Custom stop words

Custom stop words that were removed from the corpora before model training:

> *game, like, good, games, time, play, fun, way, great, little, bit, lot, pretty, feel, think, recommend, playing, things, want, different, played, worth, got, love, better, new, need, find, bad, nice, steam, know, dlc, use, hours, people, nt*[2]*, adventure, strategy, simulation, rpg, puzzle*

[2] This is probably an artifact created by stemming the falsely written *dont* (without apostrophe).