Leveraging Data From the Itch.io Online Game Distribution Platform to Help Indie Game Developers

by

Ngoc Quang Vu

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Software Engineering and Intelligent Systems

Department of Electrical and Computer Engineering
University of Alberta

© Ngoc Quang Vu, 2020

Abstract

In the game distribution world, `Steam` is often regarded as the most prominent digital platform for its many famous games made by large developers. On the other hand, the `itch.io` game distribution platform is praised for its friendliness toward small independent (indie) games developed by small teams or even a single developer. `itch.io` allows game developers to participate in online game jams (hackathons during which games are built) or publish their games at no publishing cost. In this thesis, we study game data mined from `itch.io` to help indie game developers: (1) have a higher chance of winning a game jam and (2) increase the discoverability of their games.

In the first part of the thesis, we study the game jams and their high-ranking submissions to better understand the characteristics of a popular game jam (i.e., a jam that receives many submissions) and the characteristics of high-ranking game submissions in these jams. We collected data of 1,290 past game jams and their 3,752 submissions for our analysis. We found that a quality description contributes positively to a jam's popularity and a game's ranking. Additionally, more manpower organizing a jam or developing a game increases their likelihood of being popular or high-ranking respectively. High-ranking games tend to support Windows or macOS, and belong to the "Puzzle", "Platformer", "Interactive Fiction", or "Action" genres. Finally, shorter competitive jams tend to be more popular. Our findings are useful for both future game jam organizers and participants.

In the second part of the thesis, we study an approach to increase the discoverability of the indie games hosted on `itch.io` by recommending similar indie games to players of top-selling `Steam` games. We implemented a content-based recommen-

dation technique that leverages the similarity in tags, genres, and game description between an indie game and a top-selling game using the metadata of 2,830 `itch.io` indie games and 326 top-selling `Steam` games. We then contacted the indie game developers for feedback and suggestion on our approach. We found that the majority (67.9%) of them show positive support for our idea. We analyzed the downvoted recommendations to understand the reasons and lay out the important requirements for such an indie game recommendation approach. These requirements are useful for future research and development in indie game discoverability and recommendation.

Preface

The research of this thesis has been conducted in the Analytics of Software, GAmes, and Repository Data (ASGAARD) lab led by Dr. Cor-Paul Bezemer.

Chapter 2 has been published as "Q. N. Vu and C. Bezemer, 2020. An Empirical Study of the Characteristics of Popular Game Jams and Their High-ranking Submissions on Itch.io. *In Proceedings of the 15th International Conference on the Foundations of Digital Games (FDG)* (p. 1–12)" [100]. I was responsible for the collection and cleaning of data, building and analysis of models, and manuscript composition. Dr. Bezemer was the supervisory author and was involved in concept formation and manuscript composition.

Chapter 3 has been submitted for review as "Q. N. Vu and C. Bezemer. Improving the Discoverability of Indie Games by Leveraging their Similarity to Top-Selling Games. *In Proceedings of the 43rd International Conference on Software Engineering (ICSE)*. The research conducted in this chapter received research ethics approval from the University of Alberta Research Ethics Board, Project Name "Recommendation of Indie games based on their content similarity to AAA games", Project ID "Pro00101401", July 9th, 2020. I was responsible for the collection of `itch.io` and `Steam` game data, development of recommendation algorithm, collection and analysis of game developer feedback, and manuscript composition. Dr. Bezemer was the supervisory author and was involved in concept formation and manuscript composition.

Acknowledgements

First and foremost, I would like to express my gratitude to Dr. Cor-Paul Bezemer (from the Analytics of Software, GAmes, and Repository Data (ASGAARD) lab, University of Alberta) who has given me the opportunity to undertake this research project. Dr. Bezemer's continuous guidance and support throughout the course of my project have kept me on track of the research progress. From him, I have learned a lot about research skills, from various approaches to data analysis, forming research questions, writing a scholarly paper, to presenting the research findings. Without Dr. Bezemer, my research project would not have been possible.

I would also like to thank Dr. Di Niu and Dr. Matthew Guzdial for being part of my thesis examiners.

My deepest thank to all my friends at the ASGAARD lab who helped me throughout my study, both academically and non-academically.

Last but not least, I would like to thank my family and my girlfriend who have supported me during my journey through the master's degree, especially when I encountered challenging obstacles.

Table of Contents

1 Introduction and Background **1**

1.1 Introduction 1

1.2 The Itch.io game distribution platform 4

1.2.1 Itch.io and indie games 4

1.2.2 Itch.io and online game jams 4

1.3 Thesis Outline 5

2 An Empirical Study of the Characteristics of Popular Game Jams and Their High-ranking Submissions on Itch.io **6**

2.1 Abstract 6

2.2 Introduction 7

2.3 Background and Related Work 9

2.3.1 The `itch.io` platform 9

2.3.2 Game jams 9

2.3.3 Factors affecting application success 10

2.3.4 Mining online distribution platforms 11

2.4 Methodology 12

2.4.1 Data collection 14

2.4.2 Data cleaning 14

2.4.3 Data feature encoding 15

2.5 What characterizes a popular game jam? 20

2.5.1 Motivation 20

2.5.2 Approach 20
2.5.3 Results 24
2.6 What characterizes a high-ranking submission? 28
2.6.1 Motivation 28
2.6.2 Approach 29
2.6.3 Results 31
2.7 Implications of our Findings 34
2.7.1 For jam organizers 35
2.7.2 For jam participants 35
2.7.3 For jam organizers and participants 35
2.8 Threats To Validity 36
2.8.1 Construct validity 36
2.8.2 Internal validity 37
2.8.3 External validity 37
2.9 Conclusion 38

3 Improving the Discoverability of Indie Games by Leveraging their Similarity to Top-Selling Games 39

3.1 Abstract 39
3.2 Introduction 40
3.3 Background And Related Work 43
3.3.1 An overview of the `Steam` and `itch.io` platforms 43
3.3.2 Related work 43
3.4 Methodology 46
3.4.1 Collecting game metadata 46
3.4.2 Filtering game dataset 48
3.4.3 Preprocessing textual data 48
3.4.4 Computing similarity scores 49

3.4.5 Getting indie game developer feedback 51
3.5 Preliminary study on the precision of our recommendation approach . 52
3.5.1 Motivation 52
3.5.2 Approach 52
3.5.3 Results 53
3.6 How do indie game developers feel about our recommendation approach? 55
3.6.1 Motivation 55
3.6.2 Approach 55
3.6.3 Results 55
3.7 What are the requirements of a future version of an indie game recommendation system? 57
3.7.1 Motivation 57
3.7.2 Approach 58
3.7.3 Results 62
3.8 Threats To Validity 68
3.8.1 Construct validity 69
3.8.2 Internal validity 69
3.8.3 External validity 69
3.9 Conclusion 70

4 Conclusions & Future Work **72**
4.1 Conclusion 72
4.2 Future Work 73

Bibliography **75**

List of Tables

2.1 An overview of the `itch.io` game jam dataset. 13

2.2 Jam features that are selected for our study. 17

2.3 Game features that are selected for our study. 18

2.4 Wald χ^2 values of predictors with p-values (rounded to two decimal places). 25

2.5 Wald χ^2 values of statistically significant game features with p-values (rounded to two decimal places). 30

3.1 Number of games available on different distribution platforms (as of July 5th, 2020) 44

3.2 An overview of our game dataset. 46

3.3 Component score weight of different overall similarity scores. 50

3.4 How indie game developers feel about our recommendation approach. 56

3.5 Identified reasons for bad recommendation matches. 59

3.6 Identified suggestions for our recommendation approach. 60

List of Figures

2.1 An overview of our methodology. 13

2.2 An overview of our approach to build and analyse the two models for competitive and non-competitive jams. 21

2.3 The nomograms that visualize the contribution of each feature on the competitive and non-competitive jam's popularity. The models used to build these nomograms have an AUC of 0.85 (competitive jams) and 0.74 (non-competitive jams). 23

2.4 The distributions of description length of competitive and non-competitive jams. The vertical lines are the median. Note that values have been log-transformed. 25

2.5 The distributions of jam duration of competitive and non-competitive jams. The vertical lines are the median. Note that values have been log-transformed. 27

2.6 The distributions of the number of images in game description and description length. The vertical lines are the median. Note that values are not yet log-transformed. 32

2.7 The nomogram that visualizes the contributions of game features on the game's high-ranking probability. The condensed logistic regression model used to build this nomogram has an AUC of 0.77. 33

3.1 An overview of our methodology. 47

3.2 Two simplified documents are represented as vectors of word count. Their cosine similarity score is the cosine of the angle θ between the vectors. 49
3.3 The distribution of the number of matched `Steam` games per `itch.io` game. 51
3.4 The distribution of the $AP@5$ values of the four similarity scoring methods. 54
3.5 The distribution of game genres among those that were downvoted due to genre mismatch. 63

Chapter 1

Introduction and Background

1.1 Introduction

The game industry is a large industry that is comparable in size to other global markets of other media and entertainment formats such as film, music, and pay TV [68]. The industry, with nearly 3.1 billion gamers worldwide [5], is forecast to generate $159.3 billion in revenue in 2020 [101]. While there are many large game studios, there are even more smaller independent (indie) game developers. There is no official definition of indie game studios but they are generally regarded as being small and independent of publishers [92]. Thus, indie games are games developed by small-scale developers under a limited budget.

The expansion of the game industry brought about the appearance of many online game distribution platforms. These digital platforms allow an avenue for game developers to publish their games to a wider audience. The largest of these platforms is `Steam`,[1] currently hosting more than 79,000 games. `Steam` mostly distributes large AAA games (games made by major development studios under a large development and marketing budget [17]). For indie games, `itch.io`[2] is popular store due to its optional publishing fee. As of August 14th, 2020, the `itch.io` platform currently has 281,138 games hosted [42]. All of these game stores contain a trove of interesting data that researchers can leverage to mine useful insights.

[1] https://store.steampowered.com
[2] https://itch.io

Many prior works on mining digital distribution platforms for games focus on `Steam` [8, 53, 55, 56, 88], or Nexus Mods and CurseForge [18, 50, 51, 70], distribution platforms for game mods. Because of the difference between indie games and AAA games, the knowledge from prior studies that mined `Steam` and other (mod) distribution platforms might not apply to indie games.

In this thesis, we focused on extracting useful insights from the data that is available on the `itch.io` platform. In particular, we focused on the game jams on `itch.io` and identify the characteristics of popular jams and high-ranking submissions. Additionally, we investigated an approach to improve the discoverability of indie games that are hosted on `itch.io`. We conducted the following studies:

Research Study 1: An Empirical Study of the Characteristics of Popular Game Jams and Their High-ranking Submissions on Itch.io (Chapter 2)

Motivation: Game jams are events that are similar to hackathons where developers get together to develop a playable game under a time constraint. Winning a game jam is a great bonus to a starting game developer's résumé because it is a reflection of their technical skills. Therefore, jam participants need to understand which aspects of their games need to be emphasized to have a higher chance of winning. In addition, jam hosts need to understand what to emphasize when organizing online game jams so that their jams receive more submissions. In this study, we collected data from 1,290 past online game jams hosted on `itch.io` and their 3,752 high-ranking game submissions to find out the characteristics of such jams and games.

Findings: The most important findings of this study are: quality descriptions contribute positively to both a jam's popularity and a game's high-ranking likelihood. Additionally, having been organized by more hosts or having been made by more developers increase the chance of a jam being popular and that of a game being high-ranking. Finally, multi-platform and multi-genre games are more likely to be high-ranking.

Research Study 2: Improving the Discoverability of Indie Games by

Leveraging their Similarity to Top-Selling Games (Chapter 3)

Motivation: Indie games often face difficulty in getting discovered by gamers due to their limited marketing and development budget. In addition, indie game platforms like `itch.io` often contain hundreds of thousands of indie games. Thus, the discoverability problem is a challenging problem for developers. On the other hand, popular `Steam` games often have a large player base. In this study, we present an approach to improve indie game discoverability by recommending similar indie games to players of top-selling `Steam` games. We automatically matched 2,830 `itch.io` indie games with 326 top-selling `Steam` games based on their similarity in tags, genres, and descriptions using a natural language processing technique. We conducted a large-scale user study with 195 indie developers to elicit requirements for a future recommendation system.

Findings: The majority (67.9%) of the surveyed indie game developers show positive support for our approach. We also lay out a set of requirements for future studies on such an approach. The most important ones are as follows. First, a standardized and extensive tag and genre ontology system is required to bridge `itch.io` and `Steam`. Second, a gamer's expectations should be managed when recommending indie games that are more narrow in scopes to avoid disappointment. Third, a gamer's preferences should be integrated when making recommendations. Fourth, a standardize age restriction rule between the two platforms is needed. Fifth, the recommendation tool should also recommend indie games that are the least similar to showcase their uniqueness and less popular indie games.

The findings from the first study are useful for future jam hosts when organizing an online game jam, and for future jam participants when partaking in a jam. The findings in the second study are useful for future researchers in indie game discoverability and developers of indie game recommendation systems.

1.2 The Itch.io game distribution platform

1.2.1 Itch.io and indie games

The `itch.io` game distribution platform places a strong focus on independent (indie) game developers. The platform allows them to distribute their digital content (such as games, game assets, tools, mods, and even comics, music, and books) at no publishing cost [36]. Because of this, `itch.io` attracts thousands of indie game developers who could publish fully developed games or even games that are still in the experimental phase. According to Leaf Corcoran, the creator of `itch.io`, *"We're always going to host weirder, smaller things that probably will never feel appropriate on Steam."* [36] Adding to the appealing nature of the platform, `itch.io` allows creators to customize their store pages instead of adhering to a standard layout like `Steam`. Gamers would find a much larger selection of indie games on `itch.io` than they would on `Steam`.

1.2.2 Itch.io and online game jams

Additionally, `itch.io` is a place for game jam organizers to host online game jam competitions, events where game developers gather and submit their games for ranking, similar to how hackathons work. At the time of writing, there are 4,843 past jams hosted on the platform,[3] some of which are famous competitions like the annual Game Maker's Toolkit Jam (currently `itch.io`'s biggest jam) that attract thousands of submissions while others are just small scale competitions. While many game jams are competitive (i.e., there are several prizes for winning teams), some others do not have any ranking criteria. Both types of jams offer an opportunity for game developers to showcase their projects and learn new technology in game development. Therefore, games submitted to jams range from playable prototypes to fully developed games.

The jam organizers can set multiple ranking criteria for their jams. Game entries can be judged based on their artistic style, audio, storyline, creativity, etc. There is

[3] https://itch.io/jams/past

also an overall rank that is the averaged of all other ranking criteria. A jam can last as short as a few hours to as long as several weeks. While many game entries stop being developed at the end of the jam, several others continued to be fleshed out into a complete game.

1.3 Thesis Outline

The rest of this thesis is organized as follows: Chapter 2 presents an empirical study on popular game jams and their high-ranking game submissions on `itch.io`. Chapter 3 presents our study on an indie game recommendation approach that leverages the content similarity between an indie game and a top-selling `Steam` game. Finally, Chapter 4 concludes the thesis by highlighting our key findings and discussing the potential future research directions.

Chapter 2

An Empirical Study of the Characteristics of Popular Game Jams and Their High-ranking Submissions on Itch.io

2.1 Abstract

Game jams are hackathon-like events that allow participants to develop a playable game prototype within a time limit. They foster creativity and the exchange of ideas by letting developers with different skill sets collaborate. Having a high-ranking game is a great bonus to a beginning game developer's résumé and their pursuit of a career in the game industry. However, participants often face time constraints set by jam hosts while balancing what aspects of their games should be emphasized to have the highest chance of winning. Similarly, hosts need to understand what to emphasize when organizing online jams so that their jams are more popular, in terms of submission rate. In this chapter, we study 1,290 past game jams and their 3,752 submissions on `itch.io` to understand better what makes popular jams and high-ranking games perceived well by the audience. We find that a quality description has a positive contribution to both a jam's popularity and a game's ranking. Additionally, more manpower organizing a jam or developing a game increases a jam's popularity and a game's high-ranking likelihood. High-ranking games tend to support Windows or

macOS, and belong to the "Puzzle", "Platformer", "Interactive Fiction", or "Action" genres. Also, shorter competitive jams tend to be more popular. Based on our findings, we suggest jam hosts and participants improve the description of their products and consider co-organizing or co-participating in a jam. Furthermore, jam participants should develop multi-platform multi-genre games. Finally, jam hosts should introduce a tighter time limit to increase their jam's popularity.

2.2 Introduction

Game jams are events that invite people to come together to make games in a short duration of time from several hours to several days. Traditionally, game jams are organized at a physical location. However, jam hosts sometimes face difficult logistics issues and may prefer organizing an online jam instead [40]. Some examples of online game jams are Indie Game Jams,[1] Global Game Jam,[2] and Weekly Game Jam.[3] Similarly to hackathons where developers gather and build applications in a limited time, game jams allow participants to learn, collaborate, and showcase their skills in game-making such as software development, artistic design including visuals and audio, story writing, and rapid prototyping. Therefore, winning a game jam is a great bonus to a developer's curriculum vitae because it is an indication of having the technical skills as well as the planning skill required to implement a quality software product under a time constraint. This recognition could aid the developers in their pursuit of a career in the game industry.

To the best of our knowledge, there have been no large-scale studies on game jams that explore how different factors contribute to a high-ranking game. Past studies focused on a single game jam [37, 48, 72, 74, 83]. We conducted an empirical study on game jams and games made during those jams to better understand what characterizes 1) popular jams and 2) high-ranking games.

[1] http://www.indiegamejams.com
[2] https://globalgamejam.org
[3] http://weeklygamejam.com

In this chapter, we study 1,290 game jams and 3,752 game submissions from the `itch.io`[4] online distribution platform for games. The `itch.io` platform provides a central place for organizers to host their game jam competitions. By investigating different aspects of jams with a large number of submissions and games that achieved high rankings, we could learn more about which characteristics distinguish popular jams and high-ranking submissions. In particular, we seek the answers to two Research Questions:

RQ1. What characterizes a popular game jam?

Motivation: The goal is to investigate which jam features best characterize popular online jams. We also further investigated to what extent these jam features can distinguish popular jams from non-popular jams. Our findings help jam hosts know what to prioritize to help their jams receive a high submission rate.

Findings: A better description, in terms of description length and the number of visual images, has a positive correlation with jam popularity. Additionally, more jam hosts co-organizing a jam could make it more popular. Finally, shorter competitive jams are more likely to be popular.

RQ2. What characterizes a high-ranking submission?

Motivation: The goal is to study which game features best distinguish a high-ranking submission from a low-ranking one. We also studied the extent to which these features contribute to the probability of being highly ranked. Our findings help jam participants know what to prioritize to ensure a high-ranking submission.

Findings: A better game description, in terms of description length and the number of images such as screenshots, has a positive impact on game ranking. Additionally, high-ranking games tend to be developed by more developers. Finally, games that support Windows or macOS, and are of the "Puzzle", "Platformer", "Interactive Fiction", or "Action" genres are more likely to be highly ranked.

The findings from our study are useful for future jam hosts when organizing an

[4]https://itch.io/jams/past

online game jam, and for future jam participants when partaking in a jam.

Organization of the Chapter: Section 2.3 provides background information and related work. In Section 2.4, we explain our methodology. We discuss our motivations, approaches, and results for the two Research Questions in Section 2.5 and 2.6. Section 2.7 discusses the implications of our findings. The threats to validity are described in Section 2.8. Finally, Section 2.9 concludes the chapter.

2.3 Background and Related Work

In this section, we provide a background on the `itch.io` platform and outline prior works that are related to our study.

2.3.1 The `itch.io` platform

The `itch.io` online platform allows independent digital content creators to distribute their video games. The game owners design their game page and are able to view their content's ratings, the number of followers, how their games are discovered, and downloaded or played by the audience. Additionally, `itch.io` is a place for anyone to host and participate in a game jam online. The jam hosts are free to set the requirements, the duration, whether the games should be ranked, as well as how many ranking criteria there are, and who can participate in voting. According to `itch.io`, if the jam is set to a ranked jam, the rating period starts after the jam deadline. Voters can give a score of 1 to 5 to each ranking criterion and the final rank of that criterion is based on the average score. If the host sets multiple ranking criteria, then an "Overall" rank is also calculated by averaging the results of the other criteria [39].

2.3.2 Game jams

Game jams are designed as competitions that allow participants to compete individually or as a group. Participants rapidly prototype a game, following a set theme, under

a prescribed time [2]. The competitors' backgrounds comprise of mainly programmers with different skill sets [43]. Musil et al. [65] characterized different elements of a game jam and how they play together in new product development in an engineering team. These elements include lightweight development, rapid prototyping, multidisciplinary participation, aesthetics and technology focus. Zook and Riedl [107] collected survey responses from the Global Game Jam 2013's participants to study how the development process, from design to implementation, is executed within a short period of 48 hours. They showed that participants often employ non-traditional methods of game prototyping, scope down complex ideas, ground vague ideas, and employ development iteration. Fowler et al. [22] discussed the benefits of Global Game Jams and how these benefits encourage more research and teaching activities by combining theory learning and practical experience. For an overview of studies on game jams, we refer to a study by Kultima [47].

These prior studies focused on only the Global Game Jam. Our study differs from prior work in that we studied a much larger number of jams (i.e., 1,290 jams) and we focused on the characteristics of a larger number of games (i.e., 3,752 games).

2.3.3 Factors affecting application success

Many past papers analyze mobile application success using a user-provided rating as the metric. They investigated the correlation between an app's rating and its factors. Guerrouj and Baysal [32] performed an empirical study of 474 free Android apps in 25 categories to find out how 12 app metrics across 3 categories (API quality, app-related, and user-related) impact the app's average rating. Their result shows that the number of user reviews, the app's category, and app size are the key link to an app's success. Taba et al. [94] studied the link between the UI complexity and user-perceived app quality (in terms of the number of downloads and ratings) using data from 1,292 mobile applications in 8 different categories. They found that the difference between low and high user-perceived quality is significantly affected by UI

complexity of activities. Linares-Vasquez et al. [57] analyzed how API instability and fault-proneness impact software success. Specifically, they revealed that high-rated free Android apps use APIs that are less prone to faults and change than those used by low-rated apps. Also, high app churns (i.e., more changes to the app) are related to lower user's rating, as found in a study of 154 Android apps [31]. Tian et al. [98] empirically studied the correlation between a wide range of factors and an app's success. Specifically, they analyzed 28 factors in a total of 1,492 applications from the Google Play Store and found out that high-rated apps are statistically significantly different from low-rated apps in 17 out of 28 factors. In particular, the size of an app, the number of promotional images, and the target SDK are the three most impactful factors.

These works used datasets in the mobile app domain, of which the development is largely not subjected to a time constraint similar to a game jam. Our study focuses on the range of game characteristics that jam participants should prioritize to ensure a higher chance of getting a high ranking, in the context of a jam that limits their development time.

2.3.4 Mining online distribution platforms

The studies mentioned in Section 2.3.3 focused on the Google Play[5] distribution platform. Other works focused on the Apple's App Store[6]. For a complete overview of studies on these two distribution platforms, we refer to a survey by Martin et al. [61].

Other prior works mined `Steam`[7] and related platforms: Steam Charts,[8] Steam DB,[9] Steam Spy,[10] and Steam Community.[11] Lin et al. [56] studied game reviews on

[5]https://play.google.com/store?hl=en
[6]https://www.apple.com/ios/app-store
[7]https://store.steampowered.com
[8]https://steamcharts.com
[9]https://steamdb.info
[10]https://steamspy.com
[11]https://steamcommunity.com

Steam and found that positive reviews do contain useful information for developers and that developers should pay attention to the design of the first seven hours of the game. Lin et al. [53] studied the urgent updates of the most 50 popular games on Steam and found that games that follow a frequent update strategy are more likely to have a higher proportion of urgent updates. Lin et al. [54] studied 1,182 Early Access Games to suggest that developers should use the early access model to obtain gamer's feedback on their games. Lin et al. [55] built a random forest classifier to automatically identify game bug videos. Sifa et al. [88] analyzed the playtimes of 6 million gamers on Steam to find out several archetypes of gamers. Blackburn et al. [8] studied cheaters on Steam and found that the number of cheaters in a gamer's friend network correlates with the likelihood of that gamer becoming a cheater.

Lee et al. [50] mined the Nexus Mods[12] platform to extract 9,521 game mods for the study. They found that games with official modding support have a higher median endorsement ratio for their mods and suggested that game developers and mod distribution platform developers should provide post-release support for mod developers. Poretski and Arazy [70] analyzed 45 games on Nexus Mods and found that mods can potentially increase the sales of the original games. Dey et al. [18] analyzed the most-modded games on Nexus Mods and found that untagged mods are the least popular.

Our study is the first to mine the itch.io distribution platform.

2.4 Methodology

In this section, we describe our data collection, cleaning, and feature encoding processes. Figure 2.1 gives an overview of our methodology.

[12]https://www.nexusmods.com

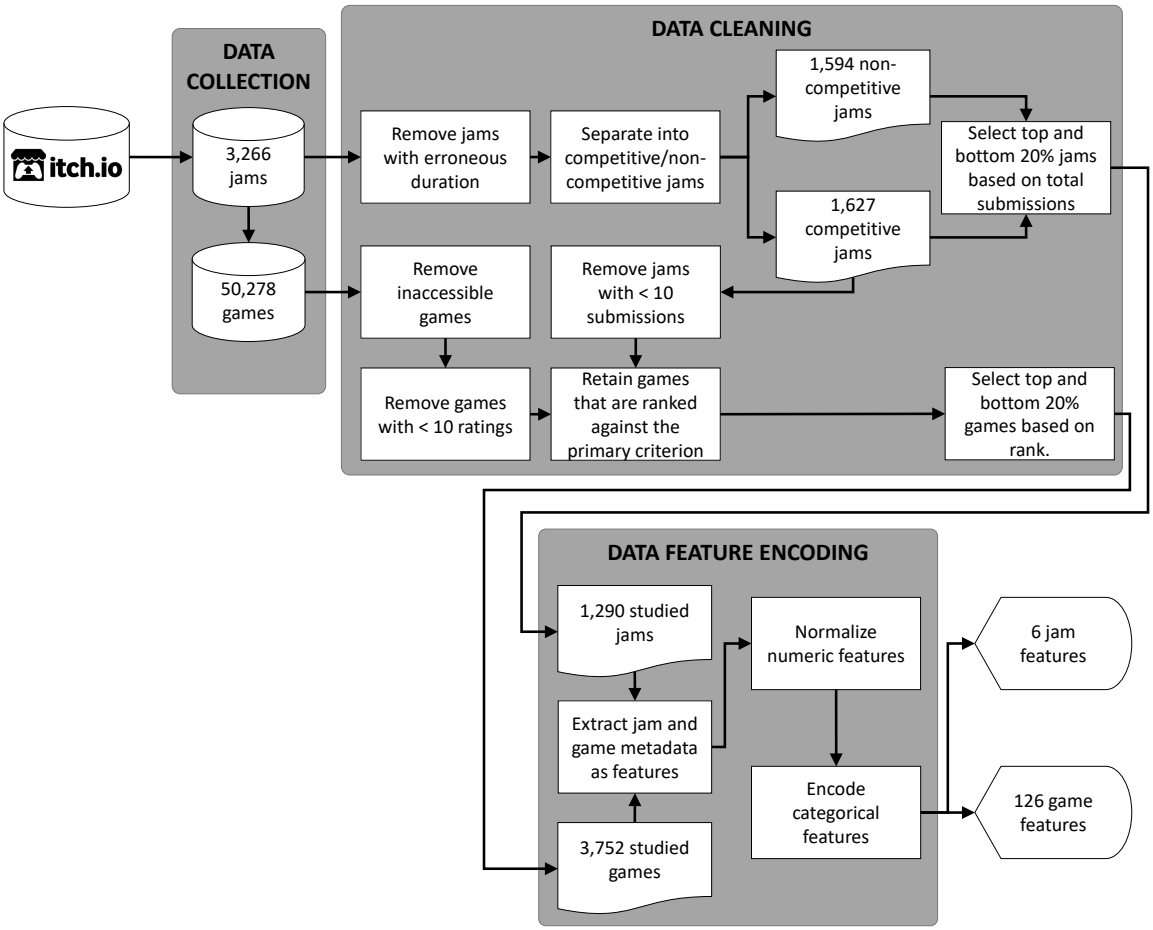


Figure 2.1: An overview of our methodology.

Table 2.1: An overview of the `itch.io` game jam dataset.

Number of total jams	3,266
Number of studied jams	1,290
Number of total games	50,278
Number of studied games	3,752
Number of studied jam features	6
Number of studied game features	126

2.4.1 Data collection

We built a customized web crawler to scrap the `itch.io` platform for past jams and games data. Our data is up-to-date as of November 13th, 2019. Table 2.1 shows an overview of our game and jam dataset.

Collecting Jams. At the time of this study, we collected 3,266 jams, the oldest one (Saltworld.net #SWJAM 2013[13]) started on September 23rd, 2013 and the latest one (IGDA Becker Chapter Halloween Jam 2019[14]) ended on October 29th, 2019. For each jam, we collected its name, URL, hosts, start and end dates, ranking criteria, jam description, and the number of submissions.

Collecting Games. We ran another customized crawler to access the jams' URLs and scrape data of all games made during those jams. We obtained 50,278 submitted game entries. The information of each game that we collected are the number of developers, supported platforms, genres, the technology used to make the game, supported inputs, supported accessibility, number of ratings, game's rankings, average playing session, license information, asset license information, and game description.

2.4.2 Data cleaning

Cleaning Jams. We removed jams of which the duration is less than one hour because we feel that any duration shorter than that would not allow participants to make a meaningful and playable game. Furthermore, some jams have a duration of a few years up to five years. We deem these jams as exceptional. Hence, we truncated the top 1% longest jams (i.e. jams that last more than 133 days). We believe any jam with a duration outside of these limits might have been erroneously set up. Next, we defined *competitive jams* as jams with at least one ranking criterion and *non-competitive jams* as those without any ranking criterion. Therefore, we separated the remaining 3,221 jams into two sets: 1,627 competitive jams and 1,594 non-competitive

[13]https://www.itch.io/jam/cgj
[14]https://www.itch.io/jam/igda-becker-chapter-halloween-jam-2019

jams.

For our first research question, for each set of competitive and non-competitive jams, we extracted the top 20% and bottom 20% of jams based on the number of submissions. We considered the top group as *popular jams* and the bottom group as *non-popular jams*. The in-between separation is to create a distinct gap between the two groups, which helps us to better model the jam's popularity. This approach has been taken previously by Tian et al. [98]. Eventually, we have 326 popular and 326 non-popular competitive jams; and 319 popular and 319 non-popular non-competitive jams.

For our second research question about game ranking, we performed further cleaning on the competitive jam dataset. We removed jams that have fewer than 10 submissions so that in a later step we could extract the top 20% and bottom 20%-ranked games and have a meaningful gap in between. We had 1,217 jams remaining.

Cleaning Games. From 50,278 games, we removed 143 games whose pages were private and not accessible. Then, of all the games that were submitted to the 1,217 jams obtained in the previous step, we removed those that have fewer than 10 ratings to make sure that the average user-rating is reliable. This is to prevent games with fake ratings (e.g. rating given by the developers themselves or their friends). The same rater bias filtering was done in prior studies [81, 98]. Next, because we wanted to study games that were ranked, we removed the games that were submitted to non-competitive jams. For competitive jams with multiple ranking criteria, we only kept games that were ranked against the "Overall" criterion. Finally, we extracted the top 20% and bottom 20%-ranked games of each jam. In total, we studied 1,876 high-ranking games and 1,876 low-ranking games.

2.4.3 Data feature encoding

Regression analysis requires independent variables to be numerical, thus all categorical features need to be encoded [73]. To do so, we converted them into indicator

variables [96]. We used this method because some categorical features can have multiple values simultaneously. For example, if a game supports both Windows and macOS, we set $platform_Windows = 1$, $platform_macOS = 1$. Next, we transformed numeric jam and game features (e.g., the number of images) using $log(x+1)$ to reduce the impact of outliers on our model. We did not consider jam prize as a jam feature for our model because we focused mainly on features that are relevant to all jam organizers who may not necessarily have access to sponsored financial resources. In total, we extracted 6 jam features and 126 game features for the study. We explained these features and their rationales in Table 2.2 and Table 2.3 respectively.

Table 2.2: Jam features that are selected for our study.

Feature	**Type**	**Description**	**Total #**	**Rationale**
num_hosts	Numeric	Number of hosts organizing the jam.	1	More hosts organizing a jam could indicate that it is a large event, which could translate to more popularity.
desc_len	Numeric	Number of characters in the jam's page description.	1	Better descriptions could help the participants understand better a jam's rules, theme, and features.
num_imgs	Numeric	Number of images or screenshots in the jam description.	1	More visual images (e.g., of past submissions) could give the audience an idea of what is expected in a jam.
num_vids	Numeric	Number of videos in the jam description.	1	Also a form of visual illustration about the jam.
duration	Numeric	Jam duration in hours.	1	The time limit could make the jam more challenging, which could affect its popularity.
num_criteria	Numeric	Number of ranking criteria (for competitive jams only).	1	More ranking criteria could make a jam more challenging, thus affecting its popularity.
Total			**6**	

Table 2.3: Game features that are selected for our study.

Feature	Type	Description	Total #	Rationale
num_devs	Numeric	Number of developers working on a game.	1	More people working on a game could result in a more polished game, leading to higher-ranking.
desc_len	Numeric	Number of characters in the game's page description.	1	Better descriptions could help the audience understand a game's rules, control, and features. The audience might rate a game higher because of this.
num_imgs	Numeric	Number of images, screenshots in the game description.	1	More visual illustrations could give the audience an idea of what a game looks like.
num_platforms	Numeric	Number of supported platforms.	1	Better platform support could translate to better ranking.
num_genres	Numeric	Number of genres the game is categorized as.	1	Games that belong to multiple genres may be preferred.
num_inputs	Numeric	Number of supported inputs devices.	1	Support for more input devices may affect a game's ranking.
num_madeWiths	Numeric	Number of tools used to develop the game.	1	Usage of more tools could give a game advantage.
*platform_ **	Boolean	Supported platform. Possible values: * = *Android, Flash, HTML5, Windows, macOS, Unity, Java, Linux*	8	The audience may have a specific preference towards a platform.
*genre_ **	Boolean	Game genre. Possible values: * = *Action, Adventure, Card.game, Educational, Fighting, Interactive.Fiction, Platformer, Puzzle, Racing, Rhythm, Role.playing, Shooter, Simulation, Sports, Strategy, Survival, Visual.novel*	17	The audience may have a specific preference towards a genre.

continued on next page

continued from previous page

Feature	Type	Description	Total #	Rationale
*input_ **	Boolean	Supported input device. Possible values: * = *Gamepad, HTC.Vive, Joy-Con, Joystick, Keyboard, Leap.Motion, Mouse, NeuroSky.Mindware, OpenSource.Virtual.Reality, Oculus.Rift, Playstation.controller, Smartphone, Touchscreen, Voice.control, Xbox.controller*	15	The audience may prefer a specific input device.
*madeWith_ **	Boolean	The technology used to develop the game. Possible values: * = *Adobe.After.Effects, Adobe.Illustrator, Adobe.Photoshop, AppGameKit, Aseprite, Audacity, Bfxr, Bitsy, Blender, Bosca ceoil, Box2D, ChipTone, Cinema.4D, Clickteam.Fusion, Clip.Studio.Paint, Construct, Corona.SDK, Defold, FL.Studio, FNA, FlashDevelop, Flickgame, GB.Studio, GIMP, GameMaker.Studio, Godot, HTC.Vive, Haxe, Impact, Inkscape, Kenney.Studio, Krita, LMMS, Logic.Pro, LÖVE, MagicaVoxel, MilkyTracker, MonoGame, Oculus.Rift, OpenFL, PICO-8, Paint.Tool.SAI, Paint.net, Phaser, Piskel, PixiJS, Processing, PuzzleScript, Pyxel.Edit, RPG.Maker, Ren'Py, Rust, SDL, SFML, Sprytile, Stencyl, Sublime.Text, Substrance.Painter_ designer, Superpowers, TIC-80, Three.js, Tiled, Twine, Unity, Unreal.Engine, ct.js, libGDX, melonJS, p5.js, pygame*	70	The usage of a specific tool could affect a game's ranking.

continued on next page

continued from previous page

Feature	Type	Description	Total #	Rationale
*avgSession_ **	Boolean	Average playing session. Possible values: * = *A.few.seconds, A.few.minutes, About.a.half-hour, About.an.hour, A.few.hours, Days.or.more*	6	The duration required to play a game could influence the audience's perception of it.
has_accessibility	Boolean	Whether the game has any accessibility support.	1	Accessibility support such as color-blind friendliness could put a game in a good light.
has_license	Boolean	Whether the game is published under a license.	1	License information could give the audience a better idea about re-using the game.
has_asset_license	Boolean	Whether the game has any asset license.	1	Some audiences could be more interested in knowing more about the game asset license.
Total			**126**	

2.5 What characterizes a popular game jam?

In this section, we present the motivation, approach, and results of Research Question 1.

2.5.1 Motivation

We studied which jam features characterize popular jams, thereby distinguishing competitive and non-competitive ones. The findings will help future hosts who want to organize an online jam and wish to increase the jam's popularity in terms of the number of submissions.

2.5.2 Approach

Figure 2.2 shows an overview of our approach to build the model. Specifically, we performed the below steps:

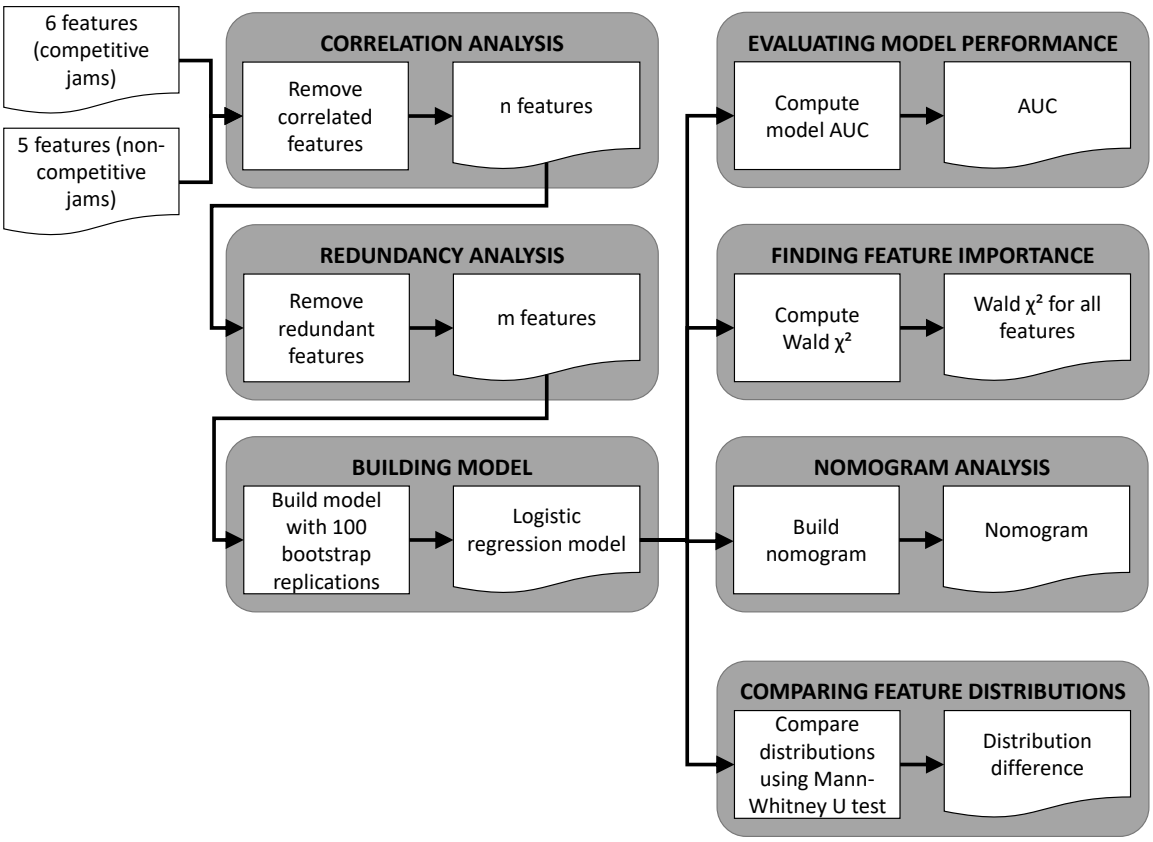


Figure 2.2: An overview of our approach to build and analyse the two models for competitive and non-competitive jams.

Correlation analysis. We needed to trim the number of variables to remove highly-correlated pairs which could affect the interpretation of the model [63]. We used *Spearman* correlation because it is resilient to features that are not normally distributed. Any pair of variables that have a *Spearman's coefficient* more than 0.7 is considered highly correlated, based on prior work that used the same threshold [56, 62, 98]. We did not observe highly correlated predictors.

Redundancy analysis. We performed a redundancy analysis to filter out redundant variables using the `Hmisc`[15] package. Redundant variables do not add information to but may affect the interpretation of the model. We did not observe redundant variables.

Model building. Using the remaining predictors after the previous step, we built

[15]https://cran.r-project.org/web/packages/Hmisc/index.html

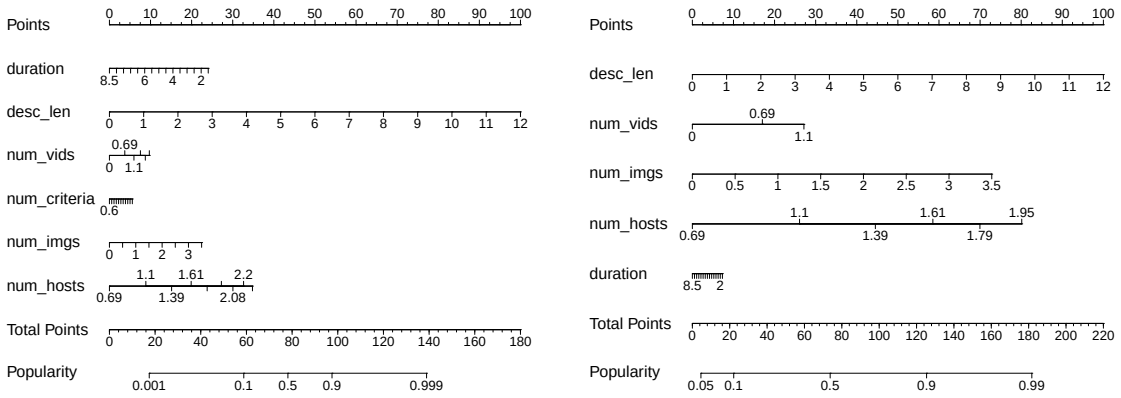
a logistic regression model, which is more interpretable than other more complex models (e.g., neural network) [64, 79]. To avoid biasing the estimated regression coefficient, we trained the model using out-of-sample bootstrap with 100 iterations. The out-of-sample bootstrap technique is performed such that for each iteration, a bootstrap sample is drawn from the original sample with replacement. Then, a model is fitted on the drawn bootstrap sample and tested on the data that are not drawn (i.e. the out-of-sample data). Finally, the mean of the sample statistics is calculated from all bootstrap samples [46].

Finding feature importance. To find out which predictors have a strong explanatory power on the response variable, we used Wald χ^2 value, computed using the `car`[16] package. A feature's importance is measured by the difference in goodness of fit between a model built with all features and a model built with the feature of interest omitted. A larger Wald χ^2 value indicates a bigger effect of the predictor [35]. Table 2.4 shows the χ^2 of the predictors of the model for competitive jams and non-competitive jams. The table also shows the associated p-values. A predictor is statistically significant if its p-value is < 0.05.

Evaluating model performance. To evaluate the performance of our models, we calculated the Area under the Receiver Operating Characteristic (ROC) curve, or AUC [33]. The AUC is commonly used as a performance metric for binary classifiers. The AUC value ranges from 0 to 1 where 1 indicates perfect prediction results, 0 indicates completely wrong predictions, and 0.5 indicates that the model's predictions are random [21]. An AUC of more than 0.7 is an indicator of a generally good model [38]. Prior Software Engineering studies [16, 52, 97] built models that have an AUC in the range 0.7 - 0.8. We observed that the AUCs of our competitive and non-competitive jam models are 0.85 and 0.74 respectively.

Nomogram analysis. We constructed each model's nomogram to visualize how a variation in each predictor can affect the outcome probability. For example, we

[16]https://cran.r-project.org/web/packages/car/index.html



(a) Competitive jams. (b) Non-competitive jams.

Figure 2.3: The nomograms that visualize the contribution of each feature on the competitive and non-competitive jam's popularity. The models used to build these nomograms have an AUC of 0.85 (competitive jams) and 0.74 (non-competitive jams).

could see how the exact number of screenshots in the description contributes to the probability of a jam being popular whereas its Wald χ^2 value only explains its importance to the model. Nomograms are regularly used to explain logistic regression models in the medical field [1, 11, 24, 86]. The nomograms for competitive and non-competitive jams are shown in Figure 2.3. For each feature, its corresponding line shows its range. The "Points" line at the top shows the magnitude of impact that each feature has based on any particular value of that feature. The "Total Points" line at the bottom shows the combined impact by all features for a given jam. The "Popularity" line shows the probability of the jam being popular based on the total points. For example, if a competitive jam has a combined point of 140 from all features, it has a probability of more than 99.9% being popular. On the other hand, if a competitive jam has less than 20 combined points, it has a probability of 0.1% being popular.

Comparing feature distributions. For features that are significant in our models, we also compared their distributions between the popular and non-popular jams. To do this, we used the one-sided Mann-Whitney U test [102] to statistically compare the distributions. The null hypothesis is that one distribution is not larger than the other. If the p-value is smaller than 0.05, we reject the null hypothesis and conclude

that one distribution is significantly larger than the other. Additionally, to understand the magnitude of this difference, we used Cliff's delta d [58] effect size. Romano et al. [77] suggested an interpretation for d as follows: negligible if $|d| \leq 0.147$; small if $0.147 < |d| \leq 0.33$; medium if $0.33 < |d| \leq 0.474$; and large if $0.474 < |d| \leq 1$. We used the package `effsize`[17] to calculate Cliff's delta.

2.5.3 Results

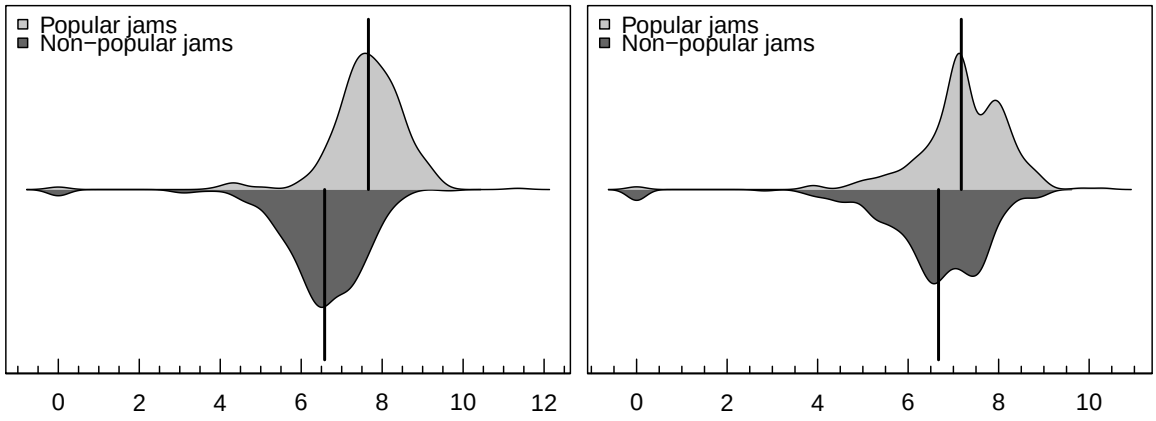
Jams that have a better description on their pages are more likely to be popular. Table 2.4 shows the sorted Wald χ^2 values of the features of our models. Table 2.4 shows that the *desc_len* feature has the most explanatory power (59.24) in competitive jams while it has the third most explanatory power (16.40) in non-competitive jams. Furthermore, the nomograms in Figure 2.3 shows that *desc_len* contributes the most to the probability of a jam being popular. Figure 2.4 shows the distributions of the description length of the jams. The descriptions of popular competitive jams have a median of 2,122 characters while that of non-popular competitive jams have a median of 719 characters. The Mann-Whitney U test confirms that the number of characters in the description of popular competitive jams is statistically significantly larger (p-value < 0.05), with a large Cliff's delta effect size (0.64). Similarly, in the case of non-competitive jams, the popular jams have a median description length of 1,305 characters while the non-popular jams have 789 characters. The Mann-Whitney U test also shows that the former's distribution is statistically significantly larger, with a medium Cliff's delta effect size (0.34).

Another aspect of jam description is the number of images. In both models, *num_imgs* are in the top three features with the most explanatory power, with Wald χ^2 values of 17.36 (competitive jams) and 17.02 (non-competitive jams). Figure 2.3 shows that having more images contributes positively to the probability of a jam being popular. The results of the Mann-Whitney U test show that popular jams

[17] https://www.rdocumentation.org/packages/effsize/versions/0.7.6/topics/effsize-package

Table 2.4: Wald χ^2 values of predictors with p-values (rounded to two decimal places).

Competitive jam			**Non-competitive jam**		
Feature	**Wald χ^2**	**p-value**	**Feature**	**Wald χ^2**	**p-value**
desc_len	59.24	$<$ **0.01**	*num_hosts*	47.17	$<$ **0.01**
num_hosts	18.39	$<$ **0.01**	*num_imgs*	17.02	$<$ **0.01**
num_imgs	17.36	$<$ **0.01**	*desc_len*	16.40	$<$ **0.01**
duration	16.54	$<$ **0.01**	*num_vids*	3.77	**0.05**
num_vids	2.76	0.10	*duration*	0.27	0.61
num_criteria	2.04	0.15			



(a) Competitive jams.

(b) Non-competitive jams.

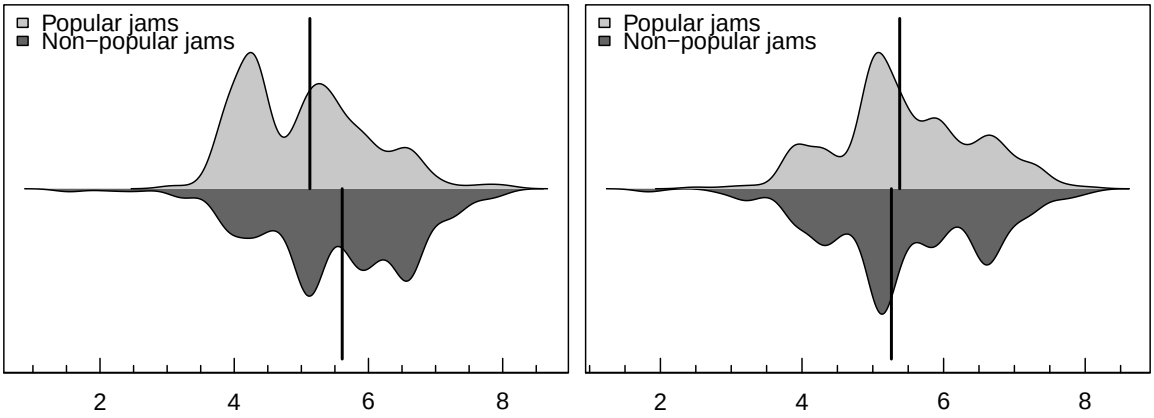
Figure 2.4: The distributions of description length of competitive and non-competitive jams. The vertical lines are the median. Note that values have been log-transformed.

have statistically significantly more images than non-popular jams (p-value < 0.05 in both competitive and non-competitive case). The corresponding effect sizes are non-negligible: 0.36 (medium, competitive jams) and 0.16 (small, non-competitive jams). Additionally, popular competitive jams have a median of one image while non-popular competitive jams have zero. Further inspection shows that, of all the studied competitive jams that have at least one image in their descriptions, 72.6% are popular jams. In the case of non-competitive jams, both popular and non-popular jams have the same median number of images, but popular non-competitive jams have more images on average. Among the studied non-competitive jams that have at least one image, 67.4% are popular.

The final aspect of jam description is the number of videos. Table 2.4 shows that *num_vids* has little explanatory power and is not a statistically significant predictor of the models. However, the Mann-Whitney U test still shows that popular jams have statistically significantly more videos in their description but with a negligible effect size (0.10) in competitive jams and a small effect size (0.16) in non-competitive jams.

Jams that are organized by more than one host are more likely to be popular. The *num_hosts* feature is the most (Wald $\chi^2 = 47.17$) and second-most (Wald $\chi^2 = 18.39$) important predictor in the non-competitive jam and competitive jam model respectively, as shown in Table 2.4. Nomogram analysis also shows that having been organized by more hosts leads to a higher probability of being more popular. The Mann-Whitney U test confirms that both popular competitive jams and popular non-competitive jams have been organized by statistically significantly (p-values < 0.05) more hosts than their non-popular counterparts, with a small effect size of 0.27 (non-competitive jams) and 0.24 (competitive jams). We further investigated the studied competitive jams that were organized by more than one host and found that 82.6% are popular. Similarly, 77.3% of the studied non-competitive jams that were organized by more than one host are popular jams.

Competitive jams are more likely to be popular if they are shorter. Ta-



(a) Competitive jams.

(b) Non-competitive jams.

Figure 2.5: The distributions of jam duration of competitive and non-competitive jams. The vertical lines are the median. Note that values have been log-transformed.

ble 2.4 shows that *duration* is the fourth most important predictor (Wald $\chi^2 = 16.54$) in the competitive jam model. However, in the non-competitive jam model, the jam duration is not statistically significant, with a p-value of 0.61. The nomogram of the competitive jam model in Figure 2.3a shows that *duration* has an inverse relationship with jam popularity. In particular, having a longer duration leads to a smaller contribution to the probability of a competitive jam being popular. Figure 2.5a and 2.5b show the distributions of jam duration in both types of jams. In the case of competitive jams, the popular ones have a median duration of 167.5 hours (nearly 7 days) while the non-popular ones have a median of 271.9 hours (11.3 days). Popular competitive jams last statistically significantly shorter than non-popular competitive jams, with a small Cliff's delta effect size of 0.28. In the case of non-competitive jams, the popular jams have a longer median duration (216.0 hours versus 192.0 hours). However, the Mann-Whitney U test shows that the difference is not statistically significant (p-value = 0.97).

The number of ranking criteria is not correlated with jam popularity. As shown in Table 2.4, the *num_ criteria* feature is the least important predictor among all in the competitive jam model, with a Wald χ^2 value of 2.04. Additionally, its

p-value of 0.15 shows that it is not a statistically significant predictor of popularity. In our sample of competitive jams, the Game Maker's Toolkit Jam 2018[18] is the third most popular jam with 1,024 submissions but has only one ranking criterion (i.e., "Design"). Furthermore, comparing the popularity, in terms of the number of submissions, between competitive jams and non-competitive jams, we found that the difference is not statistically significant based on the Mann-Whitney U test result.

Summary

Jams that have a better description, in terms of the number of images and description length, are more likely to be popular. Popular jams tend to be organized by more hosts. Additionally, shorter competitive jams tend to be popular. Finally, the number of ranking criteria in a competitive jam does not affect its popularity.

2.6 What characterizes a high-ranking submission?

In this section, we present the motivation, approach, and results of Research Question 2.

2.6.1 Motivation

We studied which game features differentiate a high-ranking game from a low-ranking game so that jam participants could understand their difference. Furthermore, some features could have a stronger impact on the outcome than others. Therefore, we would also analyze the extent to which these features affect rankings. The results will help future game jam participants understand what are the important characteristics to focus on, to improve the probability of their games achieving a high ranking.

[18] https://www.itch.io/jam/gmtk-2018

2.6.2 Approach

Our process to build and analyze the model follows the same steps as in Research Question 1:

Correlation analysis. We observed that there are 4 pairs of correlated predictors among 126 features: (1) *input_HTC.Vive* and *input_OSVR.Open.Source.Virtual.Reality*, (2) *madeWith_Logic.Pro* and *madeWith_ChipTone*, (3) *input_NeuroSky.Mindwave* and *input_HTC.Vive*, and (4) *num_inputs* and *input_Keyboard*. Their pairwise correlations are larger than 0.7. We removed *input_NeuroSky.Mindwave*, *madeWith_Logic.Pro*, *input_OSVR.Open.Source.Virtual.Reality*, and *num_inputs*.

Redundancy analysis. The analysis shows three redundant features: *num_madeWiths*, *num_platforms*, and *num_genres*. After removing them, we had 119 features left.

Model building. We built a logistic regression model using the remaining predictors. Similarly to Section 2.5, we built the model with 100 out-of-sample bootstrap iterations.

Finding feature importance. We computed the feature's Wald χ^2 value to find the degree of importance to the model. There are 18 statistically significant predictors out of 119 predictors.

Evaluating model performance. We evaluated the model performance by computing the AUC and comparing against the threshold of 0.7. We observed that our model's AUC is 0.74, which confirms that the model built from our chosen set of features can distinguish high-ranking from low-ranking games.

Nomogram analysis. To analyze the exact contribution of each feature to the probability of a game being highly ranked, we again performed nomogram analysis. However, to avoid a huge diagram, we built a condensed nomogram based on a simplified logistic regression model using only the 18 statistically significant features, with 100 bootstrap replications. The AUC of this model is 0.77.

Table 2.5: Wald χ^2 values of statistically significant game features with p-values (rounded to two decimal places).

Feature	**Wald χ^2**	**p-value**
num_imgs	223.64	< 0.01
genre_Puzzle	38.90	< 0.01
desc_len	28.11	< 0.01
num_devs	26.08	< 0.01
genre_Platformer	18.12	< 0.01
platform_Windows	15.87	< 0.01
platform_HTML5	14.66	< 0.01
madeWith_PICO.8	11.19	< 0.01
avgSession_A.few.minutes	7.83	0.01
genre_Interactive.Fiction	7.56	0.01
madeWith_GIMP	6.96	0.01
platform_Android	6.44	0.01
platform_macOS	5.94	0.01
genre_Action	5.42	0.02
madeWith_GameMaker.Studio	4.92	0.03
has_asset_license	4.76	0.03
madeWith_Paint.net	4.66	0.03
madeWith_MonoGame	4.58	0.03

Comparing feature distributions. We analyzed the feature-wise differences between high-ranking games and low-ranking games to understand the statistical significance. We again used the Mann-Whitney U test, accompanied by the Cliff's delta effect size.

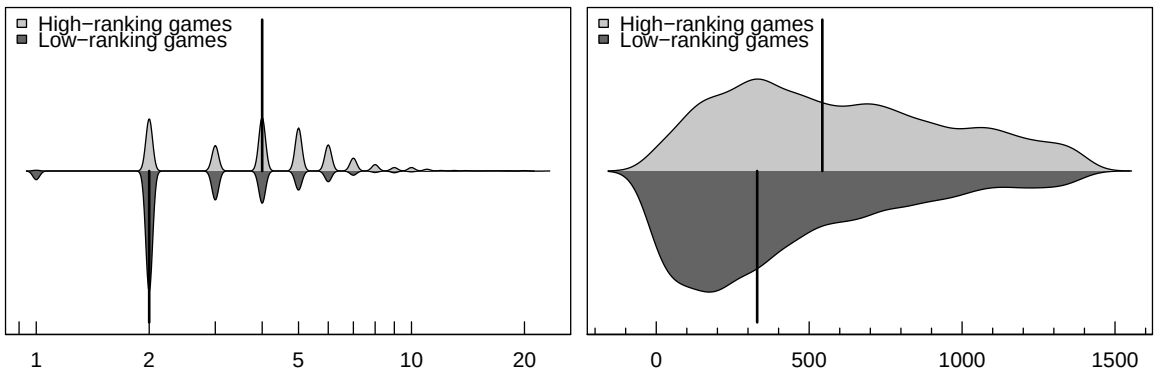
2.6.3 Results

Games that have a quality description on their pages are more likely to be highly ranked. Table 2.5 shows the features' Wald χ^2 and p-values, sorted by descending Wald χ^2. We observed that *num_imgs* has the most explanatory power (Wald χ^2 value = 223.64) on the probability of a game being highly ranked. The nomogram in Figure 2.7 shows that having more images in the game description increases the chance of the game being highly ranked. Figure 2.6a shows the distributions of the number of images in high-ranking and low-ranking games. High-ranking games have a median of four images in their description while low-ranking games have two images. The Mann-Whitney U test confirms that high-ranking games have statistically significantly more images in their description, with a medium Cliff's delta effect size of 0.42.

Additionally, the *desc_len* feature (Wald χ^2 = 28.11) is among the top three predictors with the strongest explanatory power. The descriptions of high-ranking games have a median of 543 characters, which is statistically significantly more, with a small effect size (0.29), than that of low-ranking games, which have a median of 330 characters. Figure 2.6b shows a comparison of the distributions of description length in high-ranking and low-ranking games.

The analysis results of the game description, in terms of the number of images and description length, are similar to the findings in Tian et al. [98], where the authors found that high-rated Android apps have a statistically significantly larger number of images and longer description on their Google Play's page, with a medium and small effect size respectively.

Games that belong to Puzzle, Platformer, Interactive Fiction, or Action genres are more likely to be highly ranked. The *genre_Puzzle*, *genre_Platformer*, *genre_Interactive.Fiction*, and *genre_Action* features stand at the 2nd, 5th, 10th, and 14th place respectively in the order of descending explanatory power, ac-



(a) Number of images. (b) Number of characters.

Figure 2.6: The distributions of the number of images in game description and description length. The vertical lines are the median. Note that values are not yet log-transformed.

cording to their Wald χ^2 values. Furthermore, Figure 2.7 confirms the direction of the contribution of these boolean predictors: if they have a value of 1, they contribute positively to the probability of a game being highly ranked. We further investigated the studied games in these genres and found that: 63.3% of the "Puzzle" games, 59.1% of the "Platformer" games, 51.2% of the "Interactive Fiction" games, and 51.9% of the "Action" games are high-ranking games. Overall, 56.3% of the studied games that belong to at least one of these four genres are high-ranking games.

Games that are developed by more than one developer are more likely to be highly ranked. The *num_devs* feature is the third most important predictor with a Wald χ^2 value of 26.08, as shown in Table 2.5. Figure 2.7 also shows a positive contribution to the outcome probability when there are more developers working on a game. The Mann-Whitney U test confirms that there is a statistically significant difference (p-values < 0.05) between the number of developers in high-ranking games and in low-ranking games, but the effect size is negligible (0.10). Both high-ranking games and low-ranking games have a median of one developer per game but on average, high-ranking games are made by 1.3 developers while low-ranking games are made by 1.1 developers. We investigated the studied games that were developed by more than one developer and found that 70.5% of them are highly

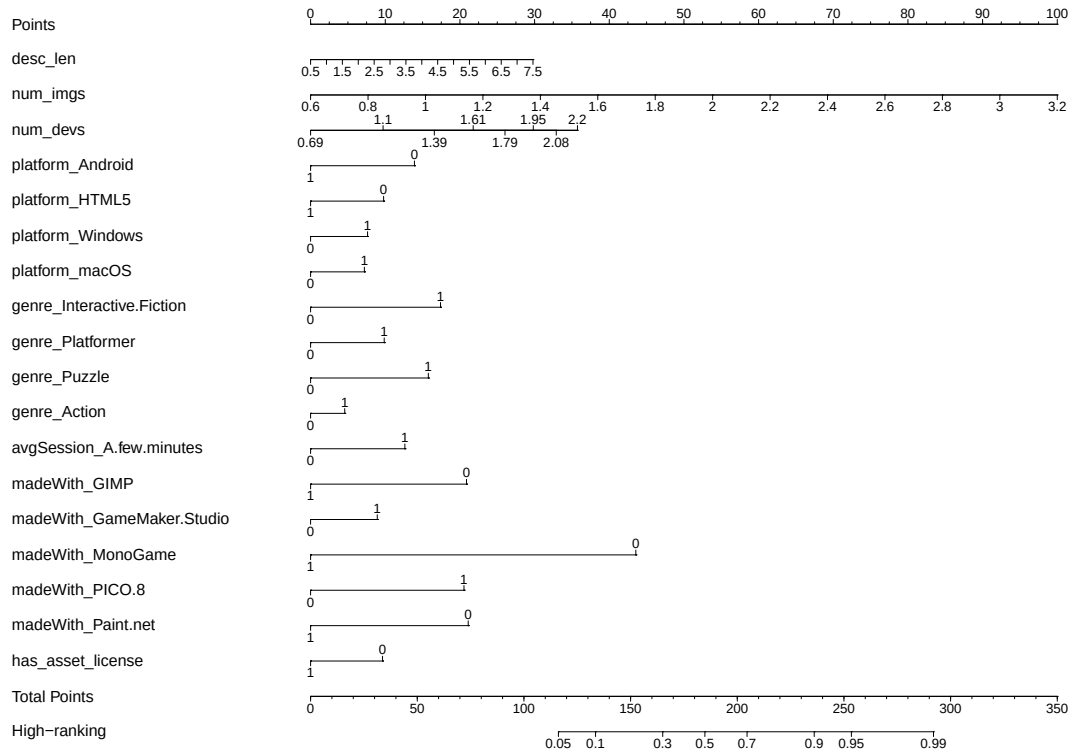


Figure 2.7: The nomogram that visualizes the contributions of game features on the game's high-ranking probability. The condensed logistic regression model used to build this nomogram has an AUC of 0.77.

ranked.

High-ranking games tend to support Windows or macOS but not HTML5 or Android. Table 2.5 shows that *platform_HTML5*, *platform_Windows*, *platform_Android*, and *platform_macOS* are statistically significant, with *platform_HTML5* and *platform_Windows* being more important than the other two, according to their Wald χ^2 values. Figure 2.7 shows that if a game supports Windows or macOS, it is more likely to be highly ranked whereas if it supports Android or HTML5, it tends to be lowly ranked. We further investigated the studied games that support Windows and macOS and found that 60.4% of the games that support Windows are highly ranked and 65.9% of the games that support macOS are highly ranked. On the other hand, 57.3% of the studied "HTML5"-supported games and 51.2% of the studied "Android"-supported games are lowly ranked. We hypothesized that Windows and

macOS tend to support better graphics and thus games that run on these platforms tend to receive higher rankings. On the other hand, browsers and mobile phones might not provide a sufficient gaming experience to players, making these games less likely to be high-ranking.

Games that are made with GameMaker Studio or PICO-8 tend to be highly ranked. Figure 2.7 shows that *madeWith_GameMaker.Studio* and *madeWith_PICO.8* have a positive contribution to the probability of a game being highly ranked. In the studied games, 60.8% of the games made with GameMaker Studio are highly ranked and 64.7% of the games made with PICO-8 are highly ranked. GameMaker Studio[19] is a cross-platform engine that allows developers to make multi-platform games using drag-and-drop visual programming language. PICO-8[20] is a game-making and game-playing console that supports a 16-color 128x128 pixel interface thus is used for developing small computer games. On the other hand, games that are made with GIMP, MonoGame, and Paint.net tend to be lowly ranked. Additionally, *madeWith_MonoGame* shows the biggest impact on the probability of a game being lowly ranked.

Summary

High-ranking games tend to exhibit these characteristics: having a quality description; belonging to either the Puzzle, Platformer, Interactive Fiction, or Action genre; supporting Windows or macOS; having been developed by GameMaker Studio or PICO-8. Finally, having been developed by more developers makes a game more likely to be highly ranked.

2.7 Implications of our Findings

In this section, we discuss the important implications of our findings and make suggestions for future jam organizers and participants based on the results.

[19]https://www.yoyogames.com/gamemaker
[20]https://www.lexaloffle.com/pico-8.php

2.7.1 For jam organizers

Jam hosts should organize shorter competitive jams. In Section 2.5, we observed that jam duration has a negative impact on the probability of a competitive jam being popular. The longer the jam, the more likely that it receives fewer submissions. We posit that a shorter time limit could pose an attractive challenge to participants and attaining high rank in such jams would be a strong recognition of their ability. Hence, if jam hosts organize a competitive jam, they should consider instating a tighter time limit to increase the competition's popularity.

2.7.2 For jam participants

Participants should develop multi-platform and multi-genre games using suitable tools. We found in Section 2.6 that, Windows or macOS support is more prevalent in high-ranking games. Additionally, high-ranking games tend to be of the "Puzzle", "Platformer", "Interactive Fiction", or "Action" genres. Therefore, participants should consider making games that have these characteristics. Since the GameMaker Studio tool could be used to make multi-platform, multi-genre games and it also helps increase the likelihood of being highly ranked, participants should consider using this tool for a combined beneficial effect.

2.7.3 For jam organizers and participants

Quality description is an important factor that affects jam popularity and game submission ranking. In both of our research questions in Section 2.5 and 2.6, we observed that the description length and the number of screenshots play a positive role in increasing the likelihood of a jam being popular, and that of a game being highly ranked. A better description is an important characteristic of popular jams and high-ranking games. We posit that better jam descriptions including more information on regulations, prizes, themes, how games are ranked, together with accompanying visuals such as images and videos (e.g. of previous submissions) could

entice more participants to the jam. Similarly, to have a better chance of being ranked high with their games, jam participants should invest more effort in improving the descriptions of their game entries, e.g. by adding screenshots.

More effort in terms of manpower increases the likelihood of a jam being popular and a game being highly ranked. We observed in Section 2.5 that the number of hosts organizing a jam contributes positively to the probability of a jam being popular, whether the jam is competitive or non-competitive. Therefore, jam hosts could consider co-organizing the jam with other hosts so that more efforts could be invested in the jam. For example, when the organizational team of a jam consists of several members, each member can focus on a specific aspect of the jam's organization (e.g., finding sponsorships for prizes, marketing the jam, and setting up technical infrastructures).

Similarly, in Section 2.6, we found that high-ranking games tend to have been developed by more than one developer. More developers could work on different aspects of a game (e.g. graphics, sounds, level design) to target different judging criteria that all affect the primary ranking. Therefore, we suggest that participants join in teams of at least two to have a better chance of being ranked high.

2.8 Threats To Validity

In this section, we describe the limitations to our work and the threats to the validity of the study.

2.8.1 Construct validity

A threat to construct validity is that although we built models from a list of game features against the game rankings, and jam features against the number of jam submissions, there is no certain causality relationship between them as such models only allow us to show correlations. Many prior studies [31, 57, 80, 94, 98] that mined code and software repositories to find the association between an app's characteristics

and its success agreed on the same limitation. Further studies could use advanced statistical causality analysis [75] to address this issue. Furthermore, the information about the game and jam features are obtained from what is presented on `itch.io` while there could be more hidden features.

2.8.2 Internal validity

A threat to internal validity is that we studied only the top 20% and bottom 20% of the jams based on their submissions, and top 20% and bottom 20% of the games based on their ranks. However, we did this to make sure there is a distinct difference between the two groups. When selecting the games, we also made sure to select from competitive jams with at least 10 submissions. The same selection technique was used in a prior study [98].

Another threat is that jam popularity may be indicated by the number of registered participants. However, not all participating teams submit a game entry. Therefore, we instead used the number of submissions as an indication of jam popularity.

In addition, some jams are already popular (e.g., GameMaker Toolkit Jam) because of their inherent characteristics such as host popularity. Therefore, our model could be biased when including them. Hence, future studies should further investigate the differences between very large game jams (such as the GameMaker Toolkit Jam) and other jams.

2.8.3 External validity

We only studied games and jams hosted on `itch.io`. However, to the best of our knowledge, this is the only online platform that hosts a large number of jams that attract thousands of games. Further studies should investigate how our approach and methods generalize to jams hosted on other platforms.

Additionally, our study focuses only on online game jams. For jams that are organized in a physical location, further studies should be done that employ interviewing

techniques with participants and hosts to understand better the characteristics of these jams.

2.9 Conclusion

In this study, we analyzed the characteristics of 1,290 online game jams and 3,752 game submissions from the `itch.io` distribution platform. First, we investigated how six jam features influence the likelihood of a jam's popularity, in terms of the number of submissions. Second, we investigated how 126 game features influence the likelihood of a game submission being highly ranked. The important takeaways of our study are:

1. A quality description not only helps a jam increase its popularity but also makes a game submission more likely to be highly ranked.
2. More manpower efforts contribute positively to a jam's popularity and a game's high-ranking probability.
3. Multi-platform multi-genre games are more likely to be highly ranked.
4. Jam hosts could consider introducing tight time limits to make the jams more popular.

Game jam organizers can use our findings to increase the likelihood of their jams being popular. In addition, game jam participants can use our findings to understand better what features are associated with high-ranking games so they can increase the likelihood of their submissions being highly ranked.

Chapter 3

Improving the Discoverability of Indie Games by Leveraging their Similarity to Top-Selling Games

3.1 Abstract

Indie games often lack visibility as compared to top-selling games due to their limited marketing budget and the fact that there are a large number of indie games. Players of top-selling games usually like certain types of games or certain game elements such as theme, gameplay, storyline. Therefore, indie games could leverage their shared game elements with top-selling games to get discovered. In this chapter, we propose an approach to improve the discoverability of indie games by recommending similar indie games to gamers of top-selling games. We first matched 2,830 `itch.io` indie games to 326 top-selling `Steam` games. We then contacted the indie game developers for evaluation feedback and suggestions. We found that the majority of them (67.9%) show positive support for our approach. We also analyzed the reasons for bad recommendations and the suggestions by indie game developers to lay out the important requirements for such a recommendation system. The most important ones are as follows. First, a standardized and extensive tag and genre ontology system is needed to bridge the two platforms. Second, the expectations of players of top-selling games should be managed to avoid disappointment. Third, a player's preferences should

be integrated when making recommendations. Fourth, a standardized age restriction rule is needed. And fifth, the recommendation tool should also show indie games that are the least similar or less popular.

3.2 Introduction

Indie games are games that are developed by a single developer or a small team who are independent of publishers [92]. With a limited marketing budget and manpower, indie game developers often face difficulty in getting their games known to gamers. To make matters worse, popular indie game distribution platforms, such as `itch.io`, may contain hundreds of thoudsands of games. Therefore, the discoverability problem is extremely challenging for indie game developers to solve. On the contrary, some of the most popular games (which are often developed and published by large game development studios) attract millions of players.

In this chapter, we propose to improve the discoverability of indie games from `itch.io` by leveraging their similarity to top-selling games from the `Steam` platform. There have been only a few prior studies on game discoverability. Past studies [87, 104] focus on game recommendation approaches for `Steam` games, or a small number of games from unspecified or less popular game stores [3, 14]. To the best of our knowledge, our study is the first study to explore a cross-store approach to improve game discoverability that targets the indie games on the `itch.io` platform and to investigate whether indie game developers value such an approach.

In particular, we recommend indie games in which gamers are likely to be interested based on their preferred top-selling `Steam` game. First, we matched 2,830 indie games with 326 top-selling games based on their similarity in tags, genres, and game descriptions. Then, we conducted a user study with 195 indie game developers to evaluate our matching results and elicit feedback and suggestions. Finally, we studied their responses to identify the most important requirements of our proposed indie game recommendation system. To find out the precision of our approach, we conducted a

preliminary study of the following Preliminary Question (PQ):

PQ. How precise is our recommendation approach?

Motivation: We calculated several similarity scores where we assigned different weights to the similarity in tags, genres, and description keywords. We evaluated the final average precision based on the upvoted recommendations by the developers to understand which similarity component of the algorithm should be more heavily weighted.

Findings: We found that a more heavily weighted tag similarity in the overall score would give a slightly better average precision at the top five recommendations. However, the performance of our recommendation approach is still low. We also collected a ground truth dataset of 2,604 recommendations from the evaluations in our user study.

In the main study, we answer the following Research Questions (RQs):

RQ1. How do indie game developers feel about improving the discoverability of their games through our approach?

Motivation: We presented our approach to indie game developers from `itch.io` and asked for their overall thoughts. The goal is to find out the main challenges to such an approach to indie game recommendation.

Findings: The majority (67.9%) of the developers support our approach while 29.6% show a neutral sentiment. The remaining 2.5% do not support our idea. We extracted two reasons from their responses. First, the developers said that indie games are too unique to be matched by similarity. Second, players of top-selling games might be disappointed when playing indie games.

RQ2. What are the requirements of a future version of an indie game recommendation system?

Motivation: We qualitatively studied the feedback on downvoted recommendations to understand what are the main reasons for the downvotes. Additionally, we analyzed the suggestions given by indie game developers. Our goal is to lay out the

requirements for future studies on the development of indie game recommendation systems.

Findings: We consolidated 10 requirements for future studies on indie game recommendation. Most importantly, a standardized and extensive tag and genre ontology system and age restriction rule are needed. Second, a gamer's expectations should be managed when recommending indie games that are more narrow in scope. Third, a gamer's preferences should be integrated when recommending similar indie games. Finally, the recommender should also recommend the indie games that are the least similar to showcase their uniqueness and less popular indie games.

The findings from our study are useful for future studies on indie game discoverability and the development of indie game recommendation systems. The most important contributions of our study are:

1. A user study with 195 indie game developers that shows the positive support for our approach from the majority of the surveyed developers.
2. A set of important requirements for indie game recommendation systems.
3. A labelled dataset that shows the developer-perceived relevance of recommendations of `itch.io` indie games and top-selling `Steam` games [93].

The rest of this chapter is organized as follows. Section 3.3 provides background information and related work. Section 3.4 explains our methodology. Section 3.5 presents a preliminary analysis on the precision of our approach. Section 3.6 describes how indie game developers feel about our approach. Section 3.7 discusses the requirements for future studies on such an approach. The threats to validity are described in Section 3.8. Finally, Section 3.9 concludes the chapter.

3.3 Background And Related Work

3.3.1 An overview of the Steam and itch.io platforms

The Steam platform is a digital game distribution platform developed by Valve Corporation and currently has more than 79,000 games. It is considered one of the largest game distribution platforms with a peak number of concurrent players of more than 20 million [91]. Table 3.1 shows the current number of games available on Steam compared to several other game distribution platforms. The games distributed on Steam are mostly AAA games. Although there is no official definition of AAA games, they are usually regarded as games developed by major game development studios under a large development and marketing budget [17]. These games often hold a spot on Steam's top-selling list [90].

On the other hand, itch.io focuses on small independent game developers to help them distribute smaller-budgeted games. In comparison to other indie game distribution platforms, itch.io is the largest one in its class (see Table 3.1). Due to the low cost of publishing games, itch.io is more suitable for small-scale independent developers.

3.3.2 Related work

Discoverability of games. The discoverability problem is faced by both indie and non-indie games alike. Indie games usually get discovered by being shared by their gamers or through word of mouth. Some examples of this sharing mechanism are: asking friends for help (Candy Crush Saga) [76], sharing game replay snippets (Poly Bridge) [99], or sharing quirky game videos (Untitled Goose Game) [23].

To the best of our knowledge, there have been no prior published studies on how effective these sharing mechanisms are in helping to increase the discoverability of indie games. Steam Labs[1] has conducted many game discoverability experiments but

[1] https://store.steampowered.com/labs

Table 3.1: Number of games available on different distribution platforms (as of July 5th, 2020)

Type	**Platform**	**# of games**
Platforms with many AAA games	Steam	79,007
	Green Man Gaming [30]	7,189
	GamersGate [27]	6,374
	GOG.com [29]	3,992
	Direct2Drive [19]	2,454
	Epic Games [20]	362
Platforms with almost no AAA games	Itch.io	264,179
	Kongregate [45]	128,664
	Newgrounds [66]	90,651
	Game Jolt [26]	18,278

their implementation details are not available to the public and they focus on games hosted on `Steam` itself. The study that is the closest to ours is one by Kholodylo and Strauss [44]. Its authors interviewed five respondents to analyze the relevance of recommender systems from the perspective of indie game developers. Our work differs from theirs in that we focus on recommending similar indie games to players of top-selling games on two game stores and we conducted a qualitative study on a much larger number (195) of indie game developers. In addition, we provide a labelled dataset of 2,604 recommendations that shows the relevant pairwise matching between several `itch.io` and `Steam` games.

Game recommendation systems. Many prior works on game recommendation focus on in-game item recommendation ([4, 7, 12, 34, 59, 95]) or leverage the gamer profile to make recommendations. Yang and Huang [104] employed text mining to extract a gamer's personality traits from their game reviews and social messages to make a game recommendation based on the similarity between their personality

and the aggregated personality of all game reviews. In the medical field, Catalá et al. [9] presented a recommendation method that uses a user's social network information to create personalized, user-friendly games for the elderly and disabled. Chow et al. [14] proposed an approach that leverages content-based and user-based information for mobile game recommendation and evaluated it on a local game mobile platform called WePlay. Sifa et al. [87] built a game recommendation system that employs an archetypal analysis approach to group similar gamers to make recommendations based on their shared preferences and tested on 500,000 users and more than 3,000 `Steam` games. Anwar et al. [3] implemented a collaborative-filtering (CF) game recommender using game ratings of a community of gamers and those of a particular gamer to recommend new games to that gamer. Smith [89] patented a method that combines gamer's profile information (e.g., previous downloaded or purchased games, playing pattern) and their friends' preferences to recommend new games to the gamer. Pérez-Marcos et al. [69] combined content-based and CF techniques to build a hybrid game recommender that was evaluated on a dataset of 3,600 `Steam` games and 11,350 gamers. Their approach showed an improvement over the baseline approach (a modified music recommender [67]). Cheuque et al. [13] proposed three `Steam` game recommendation approaches that use factorization machines, deep learning, and CF technique on game's and gamer's features and achieved better recommendation results than a baseline CF approach.

These prior studies focus on a single game store or in-game item recommendation. Our work explores an indie game recommendation approach that is based on the cross-store content-based similarity between two distinct platforms (`itch.io` and `Steam`).

Mining game distribution platforms. Most of the prior work on mining game distribution platforms focused on `Steam`. Lin et al. conducted empirical studies of several aspects of the `Steam` platform [53–56]. For example, they showed that positive game reviews (not just negative reviews) also contain useful feedback and developers should invest in the first few hours of the game, games that release frequently tend to

Table 3.2: An overview of our game dataset.

Total # of `itch.io` games	228,052
Total # of studied `itch.io` games	2,830
Total # of top-selling `Steam` games	6,210
Total # of studied `Steam` games	326

have a higher proportion of urgent updates, developers should take advantage of the early access model to gather more positive reviews, and finally, game bug videos can be identified with high precision using a random forest classifier buit using several video features such as keyword matches in video description, tags, and title. Sifa et al. [88] studied the playtimes of 6 million `Steam` gamers to detect several archetypes of gamers. Blackburn et al. [8] studied cheaters on Steam Community and revealed that the number of cheater friends correlates with the likelihood of a gamer becoming a cheater. Another group of studies mined game mod distribution platforms (e.g., Nexus Mods and CurseForge) to study how games and their mods interact with each other and yield insightful results for both game developers and mod developers [18, 50, 51, 70]. Finally, Vu and Bezemer [100] performed an empirical study on `itch.io` game jams and their submissions and found that games with a better quality description, multi-platform and multi-genre support are more likely to be high-ranking. Our work is the first cross-store study that mines game data from both `itch.io` and `Steam`.

3.4 Methodology

In this Section, we describe our experiment setup. Fig. 3.1 gives an overview of our methodology.

3.4.1 Collecting game metadata

We developed customized web crawlers to retrieve game metadata from `itch.io` and `Steam`, taking note of the crawling rules set by both sites in their respective `robot.txt`

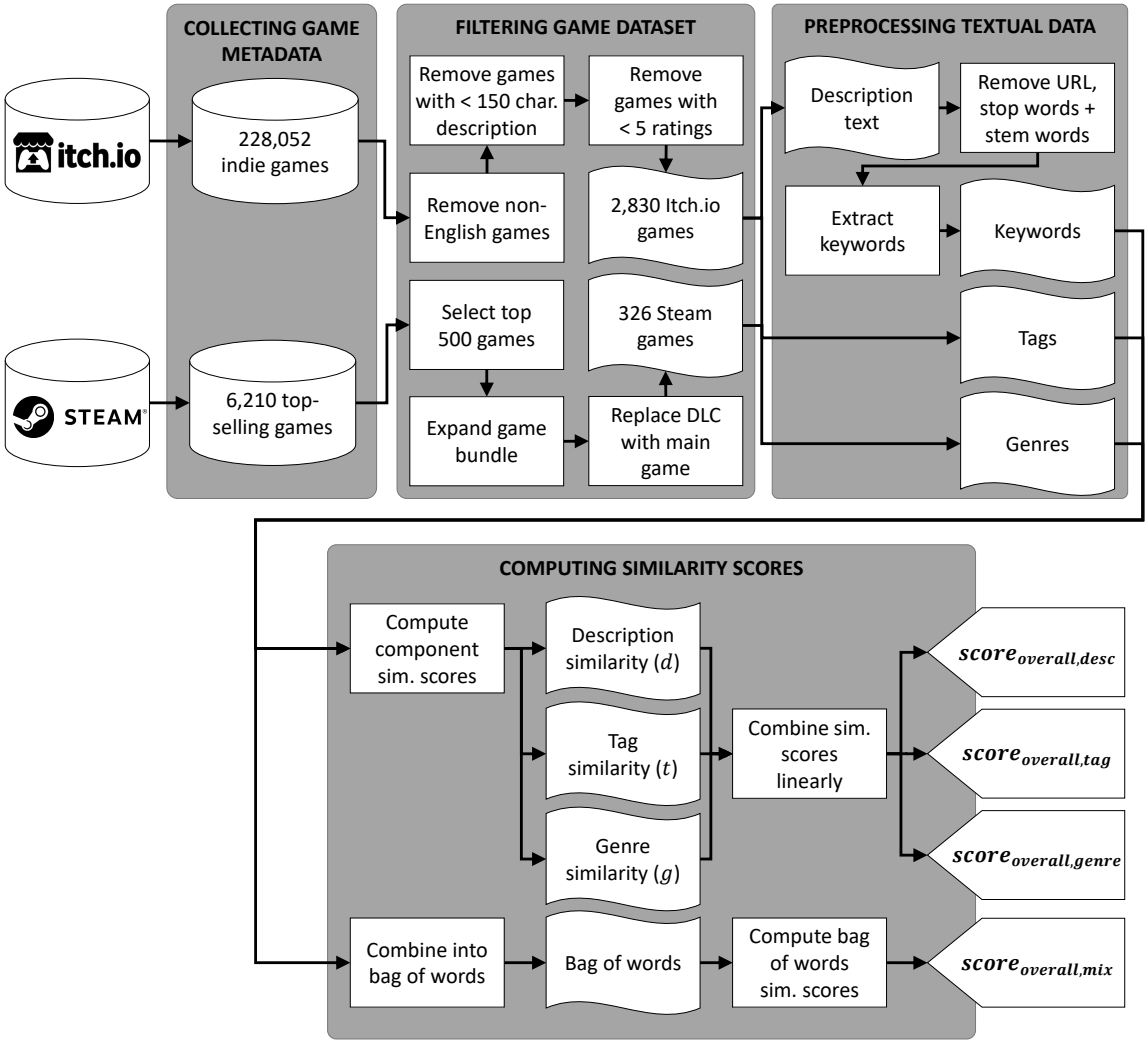


Figure 3.1: An overview of our methodology.

file. Table 3.2 shows an overview of our game dataset.

Collecting `itch.io` game metadata. We ran a crawler to retrieve a snapshot that includes the metadata of 228,052 games from `itch.io` [42] on April 15th, 2020. We collected the game name, developer(s), language, description, tag(s), genre(s), number of ratings, and URL.

Collecting `Steam` game metadata. We ran another crawler to retrieve the metadata of top-selling `Steam` games [90] on April 9th, 2020. There were 6,210 games on this list. For each of those games, we collected the game name, short description, long description, tag(s), genre(s), and URL.

3.4.2 Filtering game dataset

Filtering the `itch.io` game dataset. We removed games that are not in English based on their language metadata. We also removed games with a description length of less than 150 characters (the third quartile of the distribution of the description length). We consider the remaining games to have enough descriptive text for our text-similarity calculation. Finally, we removed games with fewer than five ratings from players; we deem these games likely to be toy or unfinished projects. Our final `itch.io` game dataset has 2,830 games.

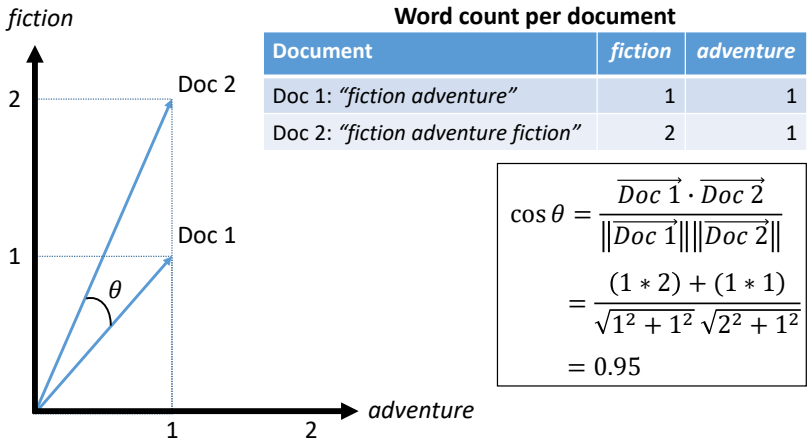
Filtering the `Steam` game dataset. Among the 6,210 top-selling `Steam` games, we kept the top 500 games; we consider these as the most popular games. We manually went through these games and checked if they are a bundle or downloadable content (DLC). If the game is a bundle of multiple games, we expanded the bundle into its component games. If it is a DLC, we replaced it with its original game. There are instances where both a game and its DLCs are in the dataset; we removed the DLCs and kept the original game. We ended up with 326 `Steam` games in our final dataset.

3.4.3 Preprocessing textual data

We preprocessed the `itch.io` game descriptions and `Steam` game main descriptions to standardize them into a common input space for our algorithms. In particular:

1. We removed HTTP links because they may be unique for each text and thus do not contribute to similarity.
2. We used the Rapid Automatic Keyword Extraction (RAKE) [78] technique to extract words that appear in candidate keyword phrases, which can represent the content of the game description concisely. In this study, we used the `rake-nltk`[2] implementation of the RAKE algorithm and used the `nltk` corpus's list of English stopwords to remove words that provide little meaning to the documents (e.g., "the", "of"). The minimum length of candidate keyword phrases

[2]https://pypi.org/project/rake-nltk



Word count per document

Document	*fiction*	*adventure*
Doc 1: *"fiction adventure"*	1	1
Doc 2: *"fiction adventure fiction"*	2	1

$$\cos\theta = \frac{\overrightarrow{Doc\ 1} \cdot \overrightarrow{Doc\ 2}}{\left\|\overrightarrow{Doc\ 1}\right\| \left\|\overrightarrow{Doc\ 2}\right\|} = \frac{(1*2)+(1*1)}{\sqrt{1^2+1^2}\sqrt{2^2+1^2}} = 0.95$$

Figure 3.2: Two simplified documents are represented as vectors of word count. Their cosine similarity score is the cosine of the angle θ between the vectors.

was set to three. Therefore, the algorithm would produce a list of words with a degree of at least three (i.e, words that co-occur with at least three words, including themselves, in all candidate keyword phrases).

3. We used the Snowball stemmer [71] to reduce the extracted words to their root form (e.g., "connected", "connection", and "connecting" are reduced to "connect").

3.4.4 Computing similarity scores

Computing component similarity scores. To assess the similarity between two games, we computed their similarity in tag, genre, and description. We used these metadata because (1) games that share one or more tags are likely to share several similar game elements described by the tags, (2) games in the same genre are likely to have similar gameplay, and (3) games that have similar descriptions are likely to have similar storylines, themes, characters, etc. For each pair of `itch.io` and `Steam` game, we computed their tag similarity score (denoted as t), genre similarity score (g), and description similarity score (d). To compute the score between two textual inputs, we converted them into numerical vector representations using the `CountVectorizer` [82]. Finally, we calculated the pairwise cosine similarity score

Table 3.3: Component score weight of different overall similarity scores.

Overall score	α	β	γ
$score_{overall,tag}$	0.50	0.25	0.25
$score_{overall,genre}$	0.25	0.50	0.25
$score_{overall,desc}$	0.25	0.25	0.50
$score_{overall,mix}$	*NA*	*NA*	*NA*

between any two games. The cosine similarity value measures the cosine of the angle between two document vectors: the smaller the angle, the greater its cosine value, the more similar the two documents are. Fig. 3.2 shows how to calculate the cosine similarity score between two sample text documents.

Computing overall similarity scores. We combined the three component scores into an overall similarity score. Since each component may contribute differently to the similarity, we assigned a weight for each score and added them linearly. Formally, we define the overall similarity score between an `itch.io` game and a `Steam` game as:

$$score_{overall} = \alpha * t + \beta * g + \gamma * d \tag{3.1}$$

where α, β, and γ are the weights of the score components. To investigate which component contributes to better similarity matching between an `itch.io` game and a `Steam` game, we computed three overall scores, each with a different component score having a weight of 0.5 and the other two component scores having an equal weight of 0.25 (the sum of all three weights is 1). We denote $score_{overall,tag}$, $score_{overall,genre}$, $score_{overall,desc}$ as the overall scores for the three scoring methods with the tag, genre, description having heavier weights respectively. Furthermore, we devised a fourth method where we put all game tags, genres, and description words together in a bag of words. We then computed a single overall similarity score (denoted as $score_{overall,mix}$)

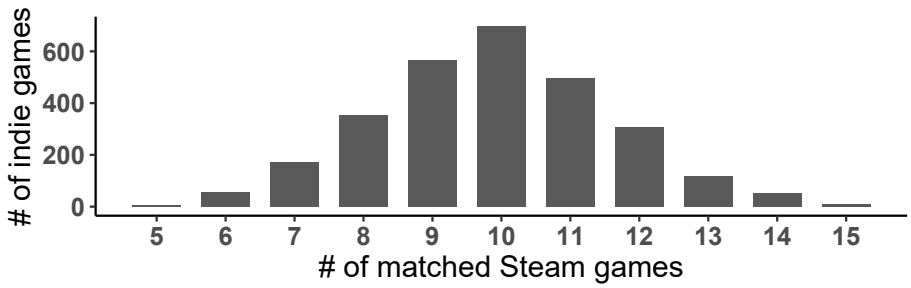


Figure 3.3: The distribution of the number of matched `Steam` games per `itch.io` game.

between the two games based on the similarity of their bags of words. Table 3.3 shows the component weights for each overall scoring method.

3.4.5 Getting indie game developer feedback

Extracting the unique set of `Steam` game matches. For each of the 2,830 indie games, we extracted the top 5 `Steam` games with the highest $score_{overall,tag}$, $score_{overall,genre}$, $score_{overall,desc}$, and $score_{overall,mix}$. Theoretically, there could be 20 `Steam` game matches for each indie game. However, because there are overlaps in the matches across the 4 overall scores, each indie game was matched with at least 5 `Steam` games and at most 15. Fig. 3.3 shows the distributions of the number of matched `Steam` games per indie game: the majority are each matched with 10 `Steam` games.

Collecting developer feedback. To evaluate the matching between an `itch.io` game and a `Steam` game, we collected feedback from the developers of the indie games in our dataset because they best understand the content of their games. We built a website[3] to display a set of matching `Steam` games for each indie game. We showed the indie game developers the reverse direction of recommendations: for each developer, we showed them their `itch.io` game and the unique set of similar `Steam` games for which their game would be recommended. We did this to collect all evaluations for each `itch.io` game. Note that the eventual implementation of our approach should

[3]See the sample screenshot in our supplementary material [93]

allow gamers to select their preferred `Steam` game to get a list of recommended indie games. The order of the matched `Steam` games was randomized when displayed to the developers to reduce the bias when evaluating the first few matches (with higher scores) versus matches further down the list. For each match, we asked the developers to give "thumbs-up" if they thought it is similar to the selected indie game, and "thumbs-down" otherwise, and elaborate on their choice. Finally, we asked them to provide their thoughts on the idea of improving the discoverability of indie games by providing suggestions to the players of top-selling games about indie games that they may enjoy. A screenshot of an example of the form to collect developer's feedback is available online [93].

We obtained the contact details of the indie game developers from their public `itch.io` profile page. These contact details include Twitter handles, email addresses, Facebook pages, and Discord servers. We managed to contact 1,526 developers and received 195 responses which equals to a response rate of 12.8%. These responses include evaluations of 2,604 `Steam` game matches of 262 `itch.io` games.

3.5 Preliminary study on the precision of our recommendation approach

3.5.1 Motivation

We studied the contributions of tags, genres, and description similarity toward good recommendations to understand which component should be more heavily weighted in the overall similarity score calculation. The findings are useful for our subsequent qualitative analysis.

3.5.2 Approach

Evaluating the precision. We used the *Average Precision at n (AP@n)* [49] metric to evaluate the precision of our recommendations. The $AP@n$ measures the number of relevant recommendations (upvotes) in the top n recommended items while also

taking into account their positions in the ranked list. We used the following variant of the formula:

$$AP@n = \frac{1}{n}\sum_{k=1}^{n}(P@k \text{ if } k^{th} \text{ item is relevant}) \tag{3.2}$$

where $P@k = \frac{r}{k}$ (k is the number of recommendations, r is the number of upvotes within the first k recommendations). Since we extracted the top five most similar games, we calculated $AP@5$ for each scoring method.

Additionally, to understand whether there is a significant difference between the $AP@5$ values of different similarity scores, we compared the distribution of the $AP@5$ values. We used the one-sided Mann-Whitney U test [102] for statistical comparison. The test has the null hypothesis that one distribution is not greater than the other. If the test returns a p-value that is less than 0.05, we conclude that one distribution is statistically larger than the other and reject the null hypothesis. To understand the magnitude of the difference, we used Cliff's delta d [58], with the thresholds suggested by Romano et al. [77]: negligible if $|d| \leq 0.147$; small if $0.147 < |d| \leq 0.33$; medium if $0.33 < |d| \leq 0.474$; and large if $0.474 < |d| \leq 1$.

3.5.3 Results

Tag similarity contributes the most to relevant recommendations. Fig. 3.4 shows the distributions of the $AP@5$ values of the four scoring methods. The recommendation method in which tag similarity is weighted heavier ($score_{overall,tag}$) has the highest median $AP@5$ at 0.17. The Mann-Whitney U test shows that this scoring method has a statistically significantly larger $AP@5$ (both p-values are 0.03) than the genre-heavy-weight ($score_{overall,genre}$) and the description-heavy-weight ($score_{overall,desc}$) method. However, the effect sizes for both comparisons are negligible (0.09). The median $AP@5$ for the latter two methods are both 0.1. Comparing between the tag-heavy-weight scoring method and the bag-of-word scoring method ($score_{overall,mix}$), there is no statistically significant difference between their $AP@5$

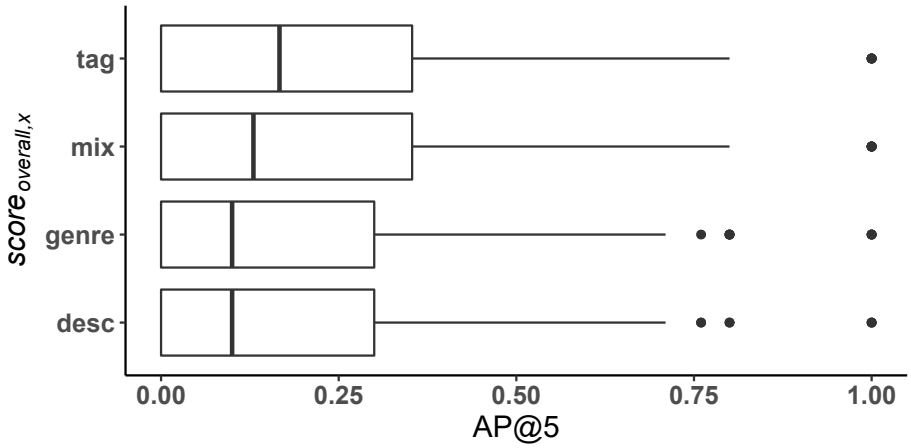


Figure 3.4: The distribution of the $AP@5$ values of the four similarity scoring methods.

values (p-value > 0.05). Finally, the bag-of-word scoring method also has a statistically significantly higher $AP@5$ than the genre-heavy- and the description-heavy-weight method, though with a negligible effect size.

All of the median $AP@5$ values remain low. However, since many indie games strive to be unique (as later found in Section 3.7), they might not have any relevant top-selling game matches (in our list, or at all). This is, therefore, an extremely difficult recommendation problem and a high average precision is hard to achieve. Additionally, to the best of our knowledge, there exists no pre-defined ground truth of relevant matchings between indie games from `itch.io` and top-selling `Steam` games. Therefore, we strived to establish an initial labelled dataset for future research in indie game discoverability and recommendation (publicly available at [93]).

Summary

Tag similarity contributes the most to relevant recommendations between an `itch.io` game and a `Steam` game. Further studies are required to optimize the weights assigned to the component similarity scores.

3.6 How do indie game developers feel about our recommendation approach?

3.6.1 Motivation

Indie games tend to lack visibility due to a low marketing budget. Therefore, we presented our recommendation approach to help increase their visibility. First and foremost it is important to understand whether indie game developers support this idea, as there may be considerations that we overlooked. Hence, first we studied whether indie game developers would support the recommendation approach.

3.6.2 Approach

Open coding the developer's sentiment in their overall feedback responses. To understand developers' sentiment about our approach, we studied the overall feedback provided by the indie game developers. We were able to extract 159 non-empty responses to the sentiment question. Next, independently, my supervisor and I manually read through these responses and categorized the sentiment of the developers into three categories: "Positive" (the developer supports the idea), "Negative" (the developer does not think it is a good idea), and "Neutral" (the developer's response does not show how they feel about the idea). Then the we compared the results and found that both agreed in the categorization of 133 responses (83.6%). There are 23 cases where one of us categorized the response as "Neutral" and the other put "Positive" or "Negative", the reason for which is that some responses are not obviously positive or negative, making it difficult to categorize them. There are only three responses where we categorized the opposite. The disagreements were resolved through discussion until a consensus was reached.

3.6.3 Results

67.9% of the indie game developers support our idea of an indie game recommendation approach. Table 3.4 shows the results, together with an example

Table 3.4: How indie game developers feel about our recommendation approach.

Sentiment	**Example**	**Count**	**Perc.**
Positive	*"A great goal. Indie games deserve more love!"*	108	67.9%
Neutral	*"Could work for some indie games, but based on the above recommendations a lot of smaller weirder indie games (which make up a fair percentage of the total) are completely their own thing and have little similarity to most AAA games."*	47	29.6%
Negative	*"Some indie games, especially gamejams such as Heatstroke often work to exist in niches that are too small for AAA to compete in so I'm am skeptical of the value of recommending these games to AAA players."*	4	2.5%

response, of our categorization process for developers' sentiment about our approach. The majority (108 out of 159) of the responses indicated that the developers think it is a good idea to help increase the discoverability of indie games based on their similarity with top-selling `Steam` games. In 47 responses (29.6%), developers did not show their support nor rejection of the idea. For example, one wrote *"Could work for some indie games, but based on the above recommendations a lot of smaller weirder indie games (which make up a fair percentage of the total) are completely their own thing and have little similarity to most AAA games."* Many developers gave suggestions on how to improve the recommendation algorithm. We discuss the suggestions in-depth in Section 3.7.

2.5% of the developers do not support our approach for indie game recommendation, due to gamer expectation management or a niche market. There are four responses (2.5%) from developers having a negative feeling about our approach. We manually read through them and extracted the following reasons:

- *Gamer expectation management:* the developers mentioned that players of top-

selling games usually have high expectations in terms of the level of polish, which is difficult to achieve for small indie game development teams. Indie games tend to be small and short therefore gamers may feel underwhelmed and not enjoyable. Prior work by Chambers et al. [10] showed that gamers are difficult to please: they have specific gameplay preferences.

- *Niche market:* the developers wrote that indie games strive to be different from other games and should not be considered as similar to top-selling games.

Although these are valid reasons, we felt that future iterations of our approach could leverage on gamers' preferences, on top of content similarity, when recommending indie games. As found in Section 3.7, many developers made such a suggestion to improve our indie game recommendation approach.

Summary

The majority (67.9%) of the developers support our idea to improve the discoverability of indie games by recommending similar indie games to top-selling `Steam` games' players. 2.5% do not support such an approach because the indie games might not meet the expectations of gamers and the indie games strive to be in a niche market.

3.7 What are the requirements of a future version of an indie game recommendation system?

3.7.1 Motivation

We received thumbs-down on some recommendations, together with the reasons given by the indie game developers. We studied these reasons to understand the shortcomings of our approach. Additionally, we analyzed the suggestions given by the developers to understand the potential improvement points to achieve higher quality recommendations. The findings will be useful for future developers and researchers of indie game recommendation systems.

```
1. Inputs = All feedback responses for matches, a list of
   categories of reason (intially an empty list)

2. For each feedback response:
   - Read the response manually for this match.

   - If the reason raised in the response matches existing
     category(-ies):
       Label the response with that/those category(-ies).

   - Else:
       Add a new category to the list of categories of reason.
       Restart the process with the new list of categories.

3. Outputs = All feedback responses labelled with categories and a
   list of categories of reason.
```

Listing 3.1: The categorization process.

3.7.2 Approach

Approach: **Open coding the feedback on downvoted recommendations.** We received 1,959 downvoted recommendations, of which 873 were accompanied by the developer's reason for downvoting. To have an overall understanding of the reasons for the bad matches, we extracted a random statistically-representative sample of 87, which has a 95% confidence level and 10% confidence interval. We manually categorized the reasons by performing an iterative process that is similar to *Open Coding* [84, 85]. Listing 3.1 shows the details of this process. My supervisor and I categorized the reasons independently and then compared the results. To calculate the agreement between the us, we used the following method. First, for each reason category, we calculated the agreement rate by counting the number of matches (we consider a match happens when we both tag or do not tag the same category for a reason) and dividing it by the sample size. Second, we took the average of the agreement rates across all categories. The final average agreement rate is 92.3%, which shows a high agreement between us. Any disagreement was discussed to reach a consensus. We extracted seven reasons for the bad recommendation matches between

Table 3.5: Identified reasons for bad recommendation matches.

Category	Description (D) - Example (E)	Count
Genre mismatch	**D:** The genres of the games are different.	41
	E: *"While the content is horror related, the genres are wildly different."*	
Gameplay mismatch	**D:** There are differences in the specific way that the games are played.	32
	E: *"Not a significantly bad match, I would just not connect the two games as their play styles are so different"*	
Look and feel mismatch	**D:** The games are different in graphics, theme, or overall mood.	22
	E: *"does not have zombies, or dark themes"*	
Target audience mismatch	**D:** The intended target audiences of both games are different.	6
	E: *"It seems to be targeted at different audience"*	
Scope mismatch	**D:** The duration of time required to play and the content size of the games are different.	3
	E: *"Dash Connect is a short casual game, while this is an open world exploration game"*	
Single/multiplayer mismatch	**D:** One game supports singleplayer or multiplayer and the other does not.	3
	E: *"We do not have multiplayer implemented yet"*	
Not similar at all/Others	**D:** Developers said that the games are entirely different, or did not give a specific reason.	21
	E: *"Not similar at all"*	

`itch.io` indie games and top-selling `Steam` games. Table 3.5 shows these categories along with their description, an example response, and the number of responses that were tagged. Since a response can be tagged with multiple reasons, the sum of all counts is larger than the sample size.

Open coding the developer's suggestions in their overall feedback responses. We used the same 159 extracted responses as in Section 3.6. Following

the same *Open Coding* approach, my supervisor and I independently read and categorized the suggestions raised in those responses. The average agreement rate was 97.7% across the categories of the suggestions. Any discrepancy was discussed until we reached a consensus. We extracted 17 categories of suggestions from the responses. Table 3.6 shows the description of the identified suggestions, along with an accompanying example response.

Table 3.6: Identified suggestions for our recommendation approach.

Category	Description (D) - Example (E)	Count
C1	**D:** The games should be matched on other similar elements such as content, theme, aesthetics.	33
	E: *"[...] the algorithm could also discover games which have a similar tone or aesthetic to the AAA games. For example, players who like Animal Crossing also like Stardew Valley. Even if the gameplay isn't necessarily similar, both games evoke a similar lighthearted, cozy feeling that scratches a similar itch."*	
C2	**D:** Developers suggest specific `Steam` game examples to be recommended.	29
	E: *"Staxel, Eco are quite similar."*	
C3	**D:** The indie game is too unique or caters to a niche target audience.	27
	E: *"A great idea, though Leximan is quite unique and the only thing I would find it comparable to is Undertale - which did not appear on the list."*	
C4	**D:** More fine-grained genres or more weight to certain genres are needed.	12
	E: *"[...] I would look into separating RPGs further, between Action RPGs, Adventure RPGs, Turn-based RPGs, etc [...]"*	
C5	**D:** The gamer's expectation should be raised right when it comes to indie games due to their usually smaller scope than that of top-selling games.	10
	E: *"[...] I think one issue in comparing small games on itch to AAA titles on steam is that the scope of games is often so different that players used to AAA scope may be disappointed in the scope of small independent games [...]"*	
C6	**D:** Indie games should only be matched with other indie games.	7
	E: *"[...] Our game is perhaps not the best to try to find a match for - all similar games we know of are indie [...]"*	

continued on next page

continued from previous page

Category	Description (D) - Example (E)	Count
C7	**D:** Developers suggest we improve our study setup such as having a similarity scale instead of just thumbs-up vs thumbs down, contacting via email instead of Twitter. **E:** *"[...] I found the thumbs up/thumbs down system a bit too restrictive, as sometimes I found myself wanting to rate something inbetween. I wish I had four options, like "similar" "slightly similar" "not very relevant" "non-relevant" [...]"*	4
C8	**D:** Developers raised a concern that some matched games are not AAA although our goal is to match with top-selling games. **E:** *"some of these games are not AAA games"*	4
C9	**D:** Developers suggested a more extensive way to query for game recommendations. **E:** *"[...] I also think it would be nice to have system that can translate description supplied by players in a natural language to queries that can filter games by features more elaborate than tags such as game mechanics, art style, etc. It could rely to gameplay videos (or trailers) to collect the needed information to index games. I usually find myself having a hard time searching for games that fit my desires even using the usual tag systems or text queries [...]"*	3
C10	**D:** This recommendation approach should be tied to an existing site and made available to the public. **E:** *"[...] The idea as a whole is phenomenal. Especially if used in either of the platforms (itch or steam) [...]"*	3
C11	**D:** The "similarity" terminology should be avoided. Instead, games should be recommended based on what the gamers might like. **E:** *"[...] The idea of improving discoverability sounds like a good one, but I think it shouldn't necessarily be correlated to big blockbuster games. It would be a bit more fair to suggest something "that the player doesn't know yet if they might like"."*	3
C12	**D:** Developers will improve the description, tag(s), and genre(s) of their indie games to help improve recommendation matching. **E:** *"[...] The issue is, many developers, including me at the time of release, try to somewhat "game" the tag system by really expanding what the "action" tag means, for example. This will properly cause issues for your fantastic site and so, I will updating the tags of my future games to reflect honestly on my games."*	1
C13	**D:** Not-Safe-For-Work (NSFW) games should not be matched with SFW games. **E:** *"[...] dont associate a sfw game with a nsfw one [...]"*	1
C14	**D:** Our recommendation should be compared to Steam recommendation.	1

continued on next page

continued from previous page

Category	Description (D) - Example (E)	Count
	E: *"[...] Perhaps interesting to compare the Steam recommendations to yours."*	
C15	**D:** There should be a limit range of popularity so that less popular games can get discovered.	1
	E: *"[...] If you want people to discover indie games, it would be better if games could only link to games within a certain range of popularity or lower than them. That would prevent users funneling upward to the biggest games, which they would probably already know about."*	
C16	**D:** Besides showing similar games, games that are entirely different should be highlighted as well.	1
	E: *"[...] it would be great if you could display also the opposite results, i.e. the indie games that do not match AAA games, those which shine for originality and creativity."*	
C17	**D:** Developers gave no suggestion in their feedback responses.	63
	E: *"A great goal. Indie games deserve more love!"*	

3.7.3 Results

In the remainder of this Section, we discuss the consolidated requirements R1–R10 that were identified from Table 3.5 and 3.6.

R1: A standardized and extensive tag and genre ontology system. Table 3.6 shows that 34 developers (C1) suggested that indie games and top-selling games should be matched on specific similar elements. These elements range from gameplay (e.g., "resource management", "platformer") to theme (e.g., "nature related") to graphics (e.g., "art style"). We observed that gameplay mismatch, and look and feel mismatch are the second and third dominant reasons for downvoted recommendations. Table 3.5 shows that 32 (36.8%) reasons for downvoting are categorized as gameplay mismatch. The feedback on these downvoted recommendations is largely about the difference in playstyle between the games. In an example match between the `You Are Undead`[4] game from `itch.io` and the `Biped`[5] game from `Steam`, the response indicated that *"The action/puzzle description kinda fits YAU's design, but*

[4]https://fabioecco.itch.io/yau
[5]https://store.steampowered.com/app/1071870/Biped

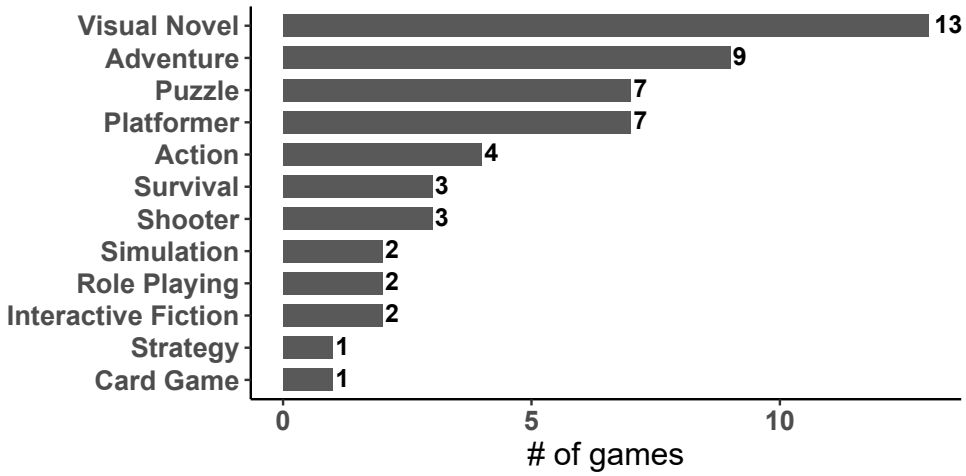


Figure 3.5: The distribution of game genres among those that were downvoted due to genre mismatch.

Biped emphasizes the co-op, 'couch play' feature. I think it wouldn't be a good recomendation. But well, it's a good guess." In the "look and feel mismatch" category, we categorized 22 responses (25.3%) (see Table 3.5). These responses mention words such as *mood*, *theme*, *atmosphere*, *tone*, and *vibe*. An example is the match between the `Tiler`[6] game and the `Warhammer 40,000: Mechanicus`[7] game, developers said that both games are strategy games but the latter one has a much heavier theme and tone while the former is simpler with flat colors. Since game tags reflect the primary game elements, having a relevant set of assigned tags that would, therefore, help improve the recommendation quality.

Similar to game tag, game genre terminology should be more fine-grained, extensively defined, and standardized between `itch.io` and `Steam` to augment the recommendation performance, as suggested by 12 developers (C4). For example, a developer suggested that their simulation game should be matched with "management simulation" games. However, both `itch.io` and `Steam` games in our dataset do not have this genre, but a common "simulation" genre. Another suggestion is to split the "Adventure" genre further into sub-genres such as "Action RPG", "Adventure RPG",

[6]https://sciman101.itch.io/tiler
[7]https://store.steampowered.com/app/673880/Warhammer_40000_Mechanicus

"Turn-based RPG".

We observed that among the reasons for downvoting a recommendation, genre mismatch is the most prevalent reason. Table 3.5 shows that 41 (47.1%) responses are categorized with "Mismatched genre". We investigated these 41 responses further to see the distribution of genres. Fig. 3.5 shows the distribution of game genres of the `itch.io` games whose matched `Steam` game received downvotes due to genre mismatch (note that a game can have multiple genres). The "Visual Novel" genre of the `itch.io` games received the largest number of downvoted `Steam` game matches due to genre mismatch. This could be because `itch.io` has a "Visual Novel" genre and no "Visual Novel" tag while `Steam` has the opposite: it has the genre but not the tag. A further check of our dataset shows that 464 (16.4%) of the studied `itch.io` games have the "Visual Novel" genre while only 6 (1.8%) of the studied top-selling `Steam` games have the "Visual Novel" tag.

The above example shows there is an inconsistency between the tags and genres used by `itch.io` and by `Steam`. In another example, one developer said that *"[...] these results were quite off, missing two of the main genres of the game like roguelike and deckbuilding, for this case [...]"*. "Roguelike" is a sub-genre of role-playing game and "Deckbuilding" is a sub-genre of card game or board game. We inspected our dataset and found that both `itch.io` and `Steam` do not have the "Roguelike" or "Deckbuilding" genre. Additionally, `itch.io` has "Roguelike", "Rogue-like", "Deck building", "Deck-builder" tags while `Steam` has "Roguelike" and "Deckbuilding" tags. Therefore, future studies should explore a more extensive tag and genre ontology system that describes the games more thoroughly and can bridge the two game stores through mapping.

Finally, there should be verification of tags assigned to a game to prevent developers from adding irrelevant tags (similar to "keyword stuffing" [41]). This is because one developer (C12) admitted that many developers "game" the tag system on `itch.io` by adding irrelevant tags to their games which would impact the similarity matching.

R2: Manage a gamer's expectations. Table 3.6 shows that 10 developers (C5) suggested gamer's expectations should be managed appropriately so that they will not feel disappointed when they come to play an indie game after playing a top-selling AAA game. We also observed that 27 developers (C3) think that indie games are too different from games in the top-selling `Steam` list. For example, one developer wrote that *"[...] Your software has the capacity to work well for indie games which are roughly analogous to AAA genres (e.g. indie fighting games to AAA fighting games, indie puzzle games to AAA puzzle games), but the independent sphere has a large number of games which simply do not fit in the AAA genre paradigm [...]"*. Additionally, seven developers (C6) suggested that indie games should only be matched with other indie games (e.g., *"The only similar experiences I've seen were coming from the indie circles [...]"*). Four developers (C8) said that some of the matched `Steam` games are not AAA games, although this could be because 132 of the 326 top-selling games in our dataset have the "indie" genre assigned.

These suggestions are consistent with the finding in Section 3.6 that a small number (2.5%) of indie game developers do not support our approach because of the difference in scope between indie games and top-selling games. However, Table 3.5 shows only three of reasons for downvoted `Steam` matches are due to the difference in scope or length of the games. This is interesting because although indie games are usually smaller in scope than a top-selling title [60] due to a limited budget, the gap between them is not the main reason for downvoted recommendations, which underlines the feasibility of our idea. Therefore, next-generation indie game recommendation systems should manage gamer's expectations well, for example by showing the recommended indie game's scope and length to gamers.

R3: Integrate a gamer's preferences when recommending similar indie games. Three developers (C11) suggested we should not use the "similarity" terminology but rather recommend games that gamers might like regardless of similarity. For example, one developer gave an example that gamers might like both `Horizon`

Zero Dawn[8] and Stardew Valley[9] despite both games having little similarity. This requirement supports the previous finding that it is difficult to match indie games with top-selling games based on similarity alone and thus, the gamer's historical preference should be leveraged for the game recommendation. We discuss several prior works in Section 3.3 that used gamer's information for game recommendation [3, 13, 14, 69, 87, 89, 104]. However, these studies used gamer's data and make game recommendations within the same platform (Steam). Additionally, it is also worth noting that data pertaining to what gamers have added to their favorite list are restricted by itch.io and Steam themselves due to privacy concerns. Past studies [54, 56] used historical game owner data from Steam Spy [25]. However, the fact that a gamer owns a game does not necessarily mean they like that game. Also, recommendation approaches that rely solely on gamer preference would face the cold-start problem: new users without any preference would not be able to receive any recommendation [28]. Therefore, future studies should explore other ways to collect preference data of gamers who play games from both itch.io and Steam platforms and combine with our similarity content-based matching approach to provide higher quality and personalized recommendations.

R4: Implement standardized age-based restriction on explicit contents. One developer (C13) suggested filtering for adult-content games from matches to a general-audience game. Both itch.io and Steam have a "NSFW" (Not-Safe-For-Work) tag for adult-content games. Therefore, an additional check is needed to verify the gamer's age before recommending games tagged with "NSFW". Additionally, while Steam has a standard age rating (i.e., ESRB), itch.io does not [15]. Because the list of tags for adult- and graphic-content can be non-exhaustive (e.g., tags such as "Gore" denote graphic violent content), there is a need to compose a common and comprehensive list of tags for such purpose.

[8] https://store.steampowered.com/app/1151640/Horizon_Zero_Dawn_Complete_Edition
[9] https://store.steampowered.com/app/413150/Stardew_Valley

R5: Allow developers to add recommendations. In Table 3.6, 29 developers (C2) wrote down the specific `Steam` games they think should be recommended for their indie games. For example, developers of the `Hardware Tycoon`[10] game recommended the `Game Dev Tycoon`[11] game due to similar gameplay and theme. It should be noted that in our study, we used a dataset of 326 top-selling `Steam` games. At the time of our data collection, the games mentioned by the developers were not on the top-selling list. However, our goal in this chapter is to show that our indie game recommendation approach can be done automatically and that it receives the majority of positive feedback. Therefore, future indie game recommendation studies should allow developers to manually add games they feel relevant to the recommendation lists (i.e., by showing a separate *staff-pick* list) or use a larger `Steam` game dataset.

R6: Allow gamers to input a description of the types of games they want to be recommended. Three developers (C9) suggested that a more flexible way to query for games should be implemented. For example, gamers should be able to give a requirement in text form, and games that match it would be recommended. Prior research by Berardi et al. [6] mined app features from app descriptions from Apple's App Store for app classification. Other prior studies investigated several approaches to search for relevant in-game moments using natural language queries [105] or visual queries [106]. These techniques could be adapted to work on gamer-supplied descriptions to identify the required matched game elements in the indie games they might like.

R7: Integrate the approach into an existing popular game distribution platform. Three developers (C10) suggested that our approach should be integrated into an existing site such as `itch.io` or `Steam` to attract a large audience.

R8: Also recommend the indie games that are the least similar to showcase their uniqueness. One developer (C16) suggested that the recommendation

[10]https://haxor1337.itch.io/hardware-tycoon
[11]https://store.steampowered.com/app/239820/Game_Dev_Tycoon

tool should also showcase indie games that do not match any top-selling games. As we discussed earlier, many developers mentioned the uniqueness of indie games, making these games difficult to be matched. Highlighting such unique games would still allow these games to benefit from the approach in this chapter. One approach to this is to gather the `itch.io` games that are the least similar to all other `Steam` games and display them as a separate list on the recommendation website.

R9: Limit the range of the recommended indie games' popularity. One developer (C15) suggested that the tool should not always recommend highly popular indie games to prevent gamers from being always drawn towards more popular games (similar to a suggestion found by Kholodylo and Strauss [44]). Therefore, future indie game recommendation tools could use metrics such as a game's rating or the number of ratings as an indicator of its popularity to make recommendations accordingly.

R10: Improve the data collection setup of the study and compare the results with `Steam`'s recommendation. Four developers (C7) suggested improving the setup for our study. In particular, they suggested the rating of recommendation should be on the scale of 1 to 10 instead of just thumbs-up and thumbs-down so that different degrees of similarity can be recorded. Another suggestion is that they prefer to be contacted via email instead of other social media means. However, the majority of `itch.io` indie game developers in our dataset do not provide public emails on their page while most have Twitter accounts. Therefore, we chose to contact them via Twitter when emails were not available. Finally, one developer (C14) suggested comparing our approach with `Steam` recommendations, although this would be difficult because the majority of `itch.io` games are not on `Steam`.

3.8 Threats To Validity

In this section, we describe the threats to the validity of the study.

3.8.1 Construct validity

Threats to *internal validity* relate to the bias of the experimenter. We relied on manual effort to categorize developer responses. These tasks are difficult to automate. Topic modeling techniques such as the latent Dirichlet allocation (LDA) require a large amount of data and rely on the use of similar game terminology across all responses while developers tend to use different terminology in their responses. A similar software engineering study [103] also analyzed survey responses manually. To mitigate this threat, the my supervisor and I categorized the responses independently and compared the results. We achieved a high level of average agreement rate and any disagreement was resolved through discussion.

3.8.2 Internal validity

We only studied games from the `itch.io` and `Steam` platforms. These two platforms are the largest of their kinds by the number of games hosted (see Table 3.1). Therefore, the games in our dataset should be representative of a large number of games which helps us identify the core improvement points for our approach. Further studies could investigate how our recommendation approach generalizes to other game distribution platforms.

Additionally, we evaluated the `itch.io` to `Steam` game matches by asking the indie game developers to rate for their games while the target implementation of the recommendation system is the other way around (i.e., allowing gamers to select their preferred `Steam` game to get recommended a list of indie games). The indie game developers may be biased about their own games. Future studies should also collect feedback from players of the `Steam` games.

3.8.3 External validity

We used open-ended questions when we asked for developer responses. Close-ended questions with a pre-defined set of answers (e.g., a pre-defined set of reasons for

bad recommendation matches) could have made it easier to aggregate the responses. However, close-ended questions are restrictive and do not provide rich answers. Since our study is the first to explore cross-store game recommendation, we felt that we needed substantial feedback from the developers to improve our approach.

Another threat is that we used the *thumbs-up vs thumbs-down* scale to measure whether a `Steam` game matches an `itch.io` game. We used this binary scale for ease of calculating the precision metrics (which need a binary input) and for separating good and bad recommendations. We found in Section 3.7 that a few developers suggested we use a scale of 1–10 instead. Future studies should explore using such a scale to evaluate recommendation approaches.

Finally, our recommendation results have fairly low average precisions. However, the goal of our study is to obtain developers' feedback on our approach and to lay out the requirements for such a recommendation system, and collect an initial set of ground truth of relevant recommendations. Therefore, we could not optimize the recommendation results before contacting the developers.

3.9 Conclusion

In this study, we proposed an approach to improve the discoverability of indie games on `itch.io` by recommending similar indie games to players of top-selling `Steam` games. First, we showed that our approach can be done automatically by calculating the similarity score between an indie game and a top-selling game. Then, we collected feedback from 195 indie game developers to help us evaluate our approach. The most important findings of our study are:

- A majority (67.9%) of indie game developers support our recommendation idea.
- It is feasible to automate the recommendation by cross-store content-based similarity matching but further work is required to optimize the results.
- A labelled dataset showing the relevant matches between `itch.io` indie games and top-selling `Steam` games [93].

We also lay out the 10 requirements for future iterations of our approach. The most important ones are:

1. An extensive standardized tag and genre ontology system.
2. Manage a gamer's expectations when recommending indie games that are more narrow in scope.
3. Integrate a gamer's preferences.
4. Implement standardized age-based restriction.
5. Also recommend indie games that are the least similar to showcase their uniqueness and less popular indie games.

Developers of indie game recommendation systems can use our findings to improve the recommendation quality. Researchers in indie game discoverability can use our findings to further extend the research direction in this field.

Chapter 4

Conclusions & Future Work

4.1 Conclusion

Prior studies on game distribution platforms that focused solely on `Steam` discovered important findings for game developers whose games are primarily AAA games. While their knowledge is useful for the development of such games, the current literature on the development of indie games and games in game jams is not much explored. In this thesis, we mined data from `itch.io`, an online game distribution platform that emphasizes on indie games and game jams. Our motivation is to help indie game developers get their games discovered and game jam participants have a higher chance of having their game submissions rank high. The contributions of the studies in this thesis are summarized as follows.

- In Chapter 2, we studied past game jams and their high-ranking submissions to understand their characteristics. We found that a high-quality description, having more co-organizers hosting a jam, or having more developers making a game contribute positively to a jam's popularity and a game's high-ranking likelihood. Additionally, jam hosts should introduce tighter jam duration to make their jams more likely to be popular. Finally, multi-platform and multi-genre games are more likely to be ranked high in a jam.

- In Chapter 3, we proposed an automatic approach to improving the discoverabil-

ity of indie games by recommending similar indie games to players of top-selling `Steam` games. Our qualitative study showed that our approach receives a majority of positive support from indie game developers from `itch.io`. We then analyzed the feedback and suggestion from the downvoted recommendations to lay out the roadmap for a future iteration of such an approach. We found the important requirements are as follows. First, a standardized and extensive tag and genre ontology system is required so that `itch.io` games and be accurately matched to top-selling `Steam` games. Second, a gamer's expectations should be managed appropriately when they are recommended to play an indie game that is more narrow in scope than the top-selling games. Third, cross-store player's preferences should be included when recommending indie games. Fourth, a standardized age restriction rule between the two platforms is needed to serve appropriate recommendations. Finally, the recommender should also suggest the indie games that are the least similar to top-selling games to showcase their uniqueness or less popupar indie games.

4.2 Future Work

We list the potential future research directions in the following list:

- **Qualitatively studying the game jam winners.** In our game jam study, we performed an empirical analysis using metadata of jams and games that are publicly available on `itch.io`. There could be more hidden characteristics that make a game high-ranking, such as developer's experience in tool usage, team dynamics in the competition. Future research should investigate these features via survey or interview with the participants.

- **Studying the characteristics of winning hackathon entries.** We studied game entries submitted to game jam competitions. Similar to a game jam, a hackathon is a time-limited competition where developers gather to make

applications to win prizes. There have been no prior large scale studies of hackathons for this purpose. Future research should investigate if our approach and findings apply to hackathons.

- **Evaluating the indie game recommendation approach through the point of view of gamers.** In our indie game recommendation study, we evaluated our approach by contacting the indie game developers for feedback. However, as the creators of their games, their feedback might be biased. Future studies should collect feedback data from gamers as well.

- **Improving the indie game recommendation algorithm.** Since our goal in the indie game recommendation study is to find out if the indie game developers think such an approach is a promising idea and to lay down the important requirements for the future development of such a recommendation system, we did not optimize our algorithm. Additionally, we did not have a pre-defined ground truth for algorithm tuning. However, we obtained an initial labeled dataset through our data collection process. Therefore, future studies could use this dataset to fine-tune the recommendation algorithm.

Bibliography

[1] V. G. Agopian, M. Harlander-Locke, A. Zarrinpar, F. M. Kaldas, D. G. Farmer, H. Yersiz, R. S. Finn, M. Tong, J. R. Hiatt, and R. W. Busuttil, "A novel prognostic nomogram accurately predicts hepatocellular carcinoma recurrence after liver transplantation: analysis of 865 consecutive liver transplant recipients," *Journal of the American College of Surgeons*, vol. 220, no. 4, pp. 416–427, 2015.

[2] M. Aho, *What is a game jam + how should you prepare*, https://www.eficode.com/blog/game-jam-preparation, (Last accessed: 2020-01-18), Sep. 2019.

[3] S. M. Anwar, T. Shahzad, Z. Sattar, R. Khan, and M. Majid, "A game recommender system using collaborative filtering (GAMBIT)," in *14th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, IEEE, 2017, pp. 328–332.

[4] V. Araujo, F. Rios, and D. Parra, "Data mining for item recommendation in MOBA games," in *Proceedings of the 13th ACM Conference on Recommender Systems*, 2019, pp. 393–397.

[5] A. Bankhurst, *Three Billion People Worldwide Now Play Video Games, New Report Shows*, (Last accessed: 2020-08-14), Aug. 2020. [Online]. Available: https://www.ign.com/articles/three-billion-people-worldwide-now-play-video-games-new-report-shows.

[6] G. Berardi, A. Esuli, T. Fagni, and F. Sebastiani, "Multi-store metadata-based supervised mobile app classification," in *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, 2015, pp. 585–588.

[7] P. Bertens, A. Guitart, P. P. Chen, and Á. Periáñez, "A Machine-Learning Item Recommendation System for Video Games," in *IEEE Conference on Computational Intelligence and Games (CIG)*, IEEE, 2018, pp. 1–4.

[8] J. Blackburn, N. Kourtellis, J. Skvoretz, M. Ripeanu, and A. Iamnitchi, "Cheating in Online Games: A Social Network Perspective," *Transactions on Internet Technology (TOIT)*, vol. 13, no. 3, p. 9, 2014.

[9] L. Catalá, V. Julián, and J.-A. Gil-Gómez, "A CBR-Based Game Recommender for Rehabilitation Videogames in Social Networks," in *International Conference on Intelligent Data Engineering and Automated Learning*, Springer, 2014, pp. 370–377.

[10] C. Chambers, W.-c. Feng, S. Sahu, and D. Saha, "Measurement-based Characterization of a Collection of On-line Games," in *Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*, 2005, pp. 1–1.

[11] S. Chen, R.-C. Nie, L.-Y. OuYang, Y.-F. Li, J. Xiang, Z.-W. Zhou, Y. Chen, and J.-S. Peng, "Nomogram analysis and external validation to predict the risk of lymph node metastasis in gastric cancer," *Oncotarget*, vol. 8, no. 7, p. 11 380, 2017.

[12] Z. Chen, T.-H. D. Nguyen, Y. Xu, C. Amato, S. Cooper, Y. Sun, and M. S. El-Nasr, "The Art of Drafting: A Team-Oriented Hero Recommendation System for Multiplayer Online Battle Arena Games," in *Proceedings of the 12th ACM Conference on Recommender Systems*, 2018, pp. 200–208.

[13] G. Cheuque, J. Guzmán, and D. Parra, "Recommender Systems for Online Video Game Platforms: the Case of STEAM," in *Companion Proceedings of The 2019 World Wide Web Conference*, 2019, pp. 763–771.

[14] A. Chow, M.-H. N. Foo, and G. Manai, "HybridRank: A Hybrid Content-Based Approach To Mobile Game Recommendations," in *CBRecSys@ RecSys*, 2014, pp. 10–13.

[15] L. Corcoran, *Lets talk about a content/age rating system for itch.io - Ideas and Feedback*, (Last accessed: 2020-08-09), Jul. 2018. [Online]. Available: https://itch.io/t/251186/lets-talk-about-a-contentage-rating-system-for-itchio.

[16] D. A. da Costa, S. L. Abebe, S. McIntosh, U. Kulesza, and A. E. Hassan, "An Empirical Study of Delays in the Integration of Addressed Issues," in *International Conference on Software Maintenance and Evolution*, IEEE, 2014, pp. 281–290.

[17] Dana, *What Are AAA Games? A Guide To Unofficial Terminology*, (Last accessed: 2020-08-12), Jul. 2020. [Online]. Available: https://geekygamingstuff.com/what-are-aaa-games/.

[18] T. Dey, J. L. Massengill, and A. Mockus, "Analysis of Popularity of Game Mods: A Case Study," in *Proceedings of the Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts*, ACM, 2016, pp. 133–139.

[19] *Direct2Drive*, (Last accessed: 2020-08-12). [Online]. Available: http://www.direct2drive.com/.

[20] *Epic Games Store*, (Last accessed: 2020-08-12). [Online]. Available: http://www.epicgames.com/.

[21] A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, *Learning from imbalanced data sets*. Springer, 2018.

[22] A. Fowler, F. Khosmood, A. Arya, and G. Lai, "The Global Game Jam for Teaching and Learning," in *Proccedings of the 4th Annual Conference on Computing and Information Technology Research and Education New Zealand*, 2013, pp. 28–34.

[23] A. Frank, *Untitled Goose Game is a funny video game about an asshole goose. Its power is in its simplicity.* (Last accessed: 2020-08-10), Oct. 2019. [Online]. Available: https://www.vox.com/culture/2019/10/10/20897212/what-is-untitled-goose-game-explained-nintendo-switch-pc-mac.

[24] N. A. Van der Gaag, J. J. Kloek, J. K. de Bakker, B. Musters, R. B. Geskus, O. R. Busch, A. Bosma, D. J. Gouma, and T. M. van Gulik, "Survival analysis and prognostic nomogram for patients undergoing resection of extrahepatic cholangiocarcinoma," *Annals of oncology*, vol. 23, no. 10, pp. 2642–2649, 2012.

[25] S. Galyonkin, *Steam Spy*, (Last accessed: 2020-07-30). [Online]. Available: http://steamspy.com/.

[26] *Game Jolt*, (Last accessed: 2020-08-12). [Online]. Available: https://gamejolt.com/.

[27] *GamersGate*, (Last accessed: 2020-08-12). [Online]. Available: http://www.gamersgate.com/.

[28] H. Gaspar, *The Cold Start Problem for Recommender Systems*, (Last accessed: 2020-08-13), Aug. 2018. [Online]. Available: https://www.yuspify.com/blog/cold-start-problem-recommender-systems/.

[29] *GOG.com*, (Last accessed: 2020-08-12). [Online]. Available: http://www.gog.com/.

[30] *Green Man Gaming*, (Last accessed: 2020-08-12). [Online]. Available: http://www.greenmangaming.com/.

[31] L. Guerrouj, S. Azad, and P. C. Rigby, "The influence of App churn on App success and Stack Overflow discussions," in *22nd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, IEEE, 2015, pp. 321–330.

[32] L. Guerrouj and O. Baysal, "Investigating the Android apps' success: An empirical study," in *24th International Conference on Program Comprehension (ICPC)*, IEEE, 2016, pp. 1–4.

[33] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*. Elsevier, 2011.

[34] L. Hanke and L. Chaimowicz, "A Recommender System for Hero Line-Ups in MOBA Games," in *Thirteenth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2017.

[35] F. E. Harrell Jr, K. L. Lee, R. M. Califf, D. B. Pryor, and R. A. Rosati, "Regression modelling strategies for improved prognostic prediction," *Statistics in medicine*, vol. 3, no. 2, pp. 143–152, 1984.

[36] P. Hernandez, *The game store that outshines Steam by staying small and weird*, (Last accessed: 2020-07-05), Nov. 2018. [Online]. Available: https://www.theverge.com/2018/11/29/18118217/itchio-steam-leaf-corcoran-pc-games-indie.

[37] X. Ho, M. Tomitsch, and T. Bednarz, "Game design inspiration in Global Game Jam," in *Proceedings of DiGRAA 2014: What is game studies in Australia?*, 2014.

[38] D. W. Hosmer Jr, S. Lemeshow, and R. X. Sturdivant, *Applied logistic regression*. John Wiley & Sons, 2013, vol. 398.

[39] *Hosting a game jam*, https://itch.io/docs/creators/game-jams, (Last accessed: 2020-01-12).

[40] *INSIGHT - Why game jams are so important*, https://www.pcgamesinsider.biz/interviews-and-opinion/67414/insight-why-game-jams-are-so-important/, (Last accessed: 2020-01-12), Jul. 2018.

[41] *Irrelevant keywords - Search Console Help*, (Last accessed: 2020-07-05). [Online]. Available: https://support.google.com/webmasters/answer/66358.

[42] Itch.io, *Top Games*, (Last accessed: 2020-08-12). [Online]. Available: https://itch.io/games.

[43] C. Kaitila, *How to Get the Most Out of a Game Jam*, https://gamedevelopment.tutsplus.com/articles/how-to-get-the-most-out-of-a-game-jam--gamedev-437, (Last accessed: 2020-01-18), Sep. 2012.

[44] M. Kholodylo and C. Strauss, "What Independent Game Developers Expect from Recommender Systems: A Qualitative Study," in *Proceedings of the 21st International Conference on Information Integration and Web-based Applications & Services*, 2019, pp. 563–570.

[45] *Kongregate*, (Last accessed: 2020-08-12). [Online]. Available: http://www.kongregate.com/.

[46] M. Kuhn and K. Johnson, *Applied predictive modeling*. Springer, 2013, vol. 26.

[47] A. Kultima, "Defining Game Jam," in *10th International Conference on the Foundations of Digital Games (FDG)*, 2015.

[48] A. Kultima and K. Alha, "Using the VNA Ideation Game at Global Game Jam," in *Proceedings of DiGRA 2011 Conference: Think Design Play*, 2011.

[49] R. R. Larson, "Introduction to Information Retrieval," *Journal of the American Society for Information Science and Technology*, vol. 61, no. 4, pp. 852–853, 2010.

[50] D. Lee, D. Lin, C.-P. Bezemer, and A. E. Hassan, "Building the Perfect Game–An Empirical Study of Game Modifications," *Empirical Software Engineering*, pp. 1–23, 2020.

[51] D. Lee, G. K. Rajbahadur, D. Lin, M. Sayagh, C.-P. Bezemer, and A. E. Hassan, "An Empirical Study of the Characteristics of Popular Minecraft Mods," *Empirical Software Engineering*, pp. 1–23, 2019.

[52] S. Lessmann, B. Baesens, C. Mues, and S. Pietsch, "Benchmarking Classification Models for Software Defect Prediction: A Proposed Framework and Novel Findings," *Transactions on Software Engineering*, vol. 34, no. 4, pp. 485–496, 2008.

[53] D. Lin, C.-P. Bezemer, and A. E. Hassan, "Studying the urgent updates of popular games on the Steam platform," *Empirical Software Engineering*, vol. 22, no. 4, pp. 2095–2126, 2017.

[54] D. Lin, C.-P. Bezemer, and A. E. Hassan, "An empirical study of early access games on the Steam platform," *Empirical Software Engineering*, vol. 23, no. 2, pp. 771–799, 2018.

[55] D. Lin, C.-P. Bezemer, and A. E. Hassan, "Identifying gameplay videos that exhibit bugs in computer games," *Empirical Software Engineering*, pp. 1–28, 2019.

[56] D. Lin, C.-P. Bezemer, Y. Zou, and A. E. Hassan, "An Empirical Study of Game Reviews on the Steam Platform," *Empirical Software Engineering*, vol. 24, no. 1, pp. 170–207, 2019.

[57] M. Linares-Vásquez, G. Bavota, C. Bernal-Cárdenas, M. Di Penta, R. Oliveto, and D. Poshyvanyk, "API change and fault proneness: a threat to the success of Android apps," in *Proceedings of the 9th joint meeting on Foundations of Software Engineering*, ACM, 2013, pp. 477–487.

[58] J. D. Long, D. Feng, and N. Cliff, "Ordinal analysis of behavioral data," *Handbook of psychology*, 2003.

[59] W. Looi, M. Dhaliwal, R. Alhajj, and J. Rokne, "Recommender System for Items in Dota 2," *IEEE Transactions on Games*, 2018.

[60] B. Lowry, *The major differences between 'indie' and 'AAA' video games*, (Last accessed: 2020-07-25), Nov. 2017. [Online]. Available: https://www.windowscentral.com/indie-vs-aaa-which-type-game-you.

[61] W. Martin, F. Sarro, Y. Jia, Y. Zhang, and M. Harman, "A Survey of App Store Analysis for Software Engineering," *Transactions on Software Engineering*, vol. 43, no. 9, pp. 817–847, 2016.

[62] S. McIntosh, Y. Kamei, B. Adams, and A. E. Hassan, "An empirical study of the impact of modern code review practices on software quality," *Empirical Software Engineering*, vol. 21, no. 5, pp. 2146–2189, 2016.

[63] H. Midi, S. K. Sarkar, and S. Rana, "Collinearity diagnostics of binary logistic regression model," *Journal of Interdisciplinary Mathematics*, vol. 13, no. 3, pp. 253–267, 2010.

[64] C. Molnar, *Interpretable Machine Learning*, 2020. [Online]. Available: \url{https://christophm.github.io/interpretable-ml-book}.

[65] J. Musil, A. Schweda, D. Winkler, and S. Biffl, "Synthesized essence: what game jams teach about prototyping of new software products," in *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 2*, ACM, 2010, pp. 183–186.

[66] *Newgrounds*, (Last accessed: 2020-08-12). [Online]. Available: http://www.newgrounds.com/.

[67] M. Pacula, *A matrix factorization algorithm for music recommendation using implicit user feedback*, 2009.

[68] E. Peckham, *Newzoo forecasts 2020 global games industry will reach $159 billion*, (Last accessed: 2020-08-12), Jun. 2020. [Online]. Available: https://techcrunch.com/2020/06/26/newzoo-forecasts-2020-global-games-industry-will-reach-159-billion/.

[69] J. Pérez-Marcos, L. Martín-Gómez, D. M. Jiménez-Bravo, V. F. López, and M. N. Moreno-García, "Hybrid system for video game recommendation based on implicit ratings and social networks," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–11, 2020.

[70] L. Poretski and O. Arazy, "Placing Value on Community Co-creations: A Study of a Video Game 'Modding' Community," in *Proceedings of the Conference on Computer Supported Cooperative Work and Social Computing*, ACM, 2017, pp. 480–491.

[71] M. F. Porter, *Snowball: A language for stemming algorithms*, 2001.

[72] J. A. Preston, J. Chastine, C. O'Donnell, T. Tseng, and B. MacIntyre, "Game Jams: Community, Motivations, and Learning among Jammers," *International Journal of Game-Based Learning (IJGBL)*, vol. 2, no. 3, pp. 51–70, 2012.

[73] C. R. Rao, H. Toutenburg, H. C. Shalabh, and M Schomaker, "Linear Models and Generalizations," *Least Squares and Alternatives (3rd edition) Springer, Berlin Heidelberg New York*, 2008.

[74] L. Reng, H. Schoenau-Fog, and L. B. Kofoed, "The Motivational Power of Game Communities–Engaged through Game Jamming," in *Proceedings of the 8th International Conference on the Foundations of Digital Games*, 2013, pp. 14–17.

[75] R. D. Retherford and M. K. Choe, *Statistical models for causal analysis*. John Wiley & Sons, 2011.

[76] J. D. Rocca, *Discoverability by Design: Rethinking How Video Games Get Noticed*, (Last accessed: 2020-08-13), Jun. 2020. [Online]. Available: https://trends.cmf-fmc.ca/embedded-discoverability-video-games.

[77] J. Romano, J. D. Kromrey, J. Coraggio, J. Skowronek, and L. Devine, "Exploring methods for evaluating group differences on the NSSE and other surveys: Are the t-test and Cohen's d indices the most appropriate choices," in *Annual meeting of the Southern Association for Institutional Research*, 2006.

[78] S. Rose, D. Engel, N. Cramer, and W. Cowley, "Automatic Keyword Extraction from Individual Documents," *Text mining: applications and theory*, vol. 1, pp. 1–20, 2010.

[79] A. Ruiz and N. Villa, "Storms prediction: Logistic regression vs random forest for unbalanced data," *arXiv preprint arXiv:0804.0650*, 2008.

[80] I. J. M. Ruiz, M. Nagappan, B. Adams, T. Berger, S. Dienst, and A. E. Hassan, "On the Relationship between the Number of Ad Libraries in an Android App and its Rating," *IEEE Software*, vol. 99, no. 1, 2014.

[81] I. J. M. Ruiz, M. Nagappan, B. Adams, T. Berger, S. Dienst, and A. E. Hassan, "Examining the Rating System Used in Mobile-App Stores," *IEEE Software*, vol. 33, no. 6, pp. 86–92, 2015.

[82] Scikit-learn, *CountVectorizer*, (Last accessed: 2020-07-25). [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html.

[83] M. J. Scott and G. Ghinea, "Promoting Game Accessibility: Experiencing an Induction on Inclusive Design Practice at the Global Games Jam," *arXiv preprint arXiv:1305.4359*, 2013.

[84] C. B. Seaman, "Qualitative Methods in Empirical Studies of Software Engineering," *IEEE Transactions on Software Engineering*, vol. 25, no. 4, pp. 557–572, 1999.

[85] C. B. Seaman, F. Shull, M. Regardie, D. Elbert, R. L. Feldmann, Y. Guo, and S. Godfrey, "Defect categorization: making use of a decade of widely varying historical data," in *Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement*, 2008, pp. 149–157.

[86] S. F. Shariat, P. I. Karakiewicz, G. Godoy, and S. P. Lerner, "Use of nomograms for predictions of outcome in patients with advanced bladder cancer," *Therapeutic advances in urology*, vol. 1, no. 1, pp. 13–26, 2009.

[87] R. Sifa, C. Bauckhage, and A. Drachen, "Archetypal Game Recommender Systems," in *Proceedings of the LWA 2014 Workshops: KDML, IR and FGWM*, 2014, pp. 45–56.

[88] R. Sifa, C. Bauckhage, and A. Drachen, "The Playtime Principle: Large-scale cross-games interest modeling," in *Conference on Computational Intelligence and Games*, IEEE, 2014, pp. 1–8.

[89] D. Smith, *Recommendation engine for electronic game shopping channel*, US Patent 8,298,087, Oct. 2012.

[90] Steam, *Top Sellers*, (Last accessed: 2020-08-12). [Online]. Available: https://store.steampowered.com/search/?filter=topsellers.

[91] *Steam: Game and Player Statistics*, (Last accessed: 2020-07-05). [Online]. Available: https://store.steampowered.com/stats/Steam-Game-and-Player-Statistics.

[92] C. Stern, *What makes a game indie: a universal definition*, (Last accessed: 2020-08-01), Apr. 2015. [Online]. Available: https://sinisterdesign.net/what-makes-a-game-indie-a-universal-definition.

[93] *Supplementary material for our paper: Labelled Dataset and Sample Screenshot*, (Last accessed: 2020-08-13). [Online]. Available: https://figshare.com/s/bed46a84774015f89b7b.

[94] S. E. S. Taba, I. Keivanloo, Y. Zou, J. Ng, and T. Ng, "An Exploratory Study on the Relation between User Interface Complexity and the Perceived Quality," in *International Conference on Web Engineering*, Springer, 2014, pp. 370–379.

[95] Z. Tao, M. Cheung, J. She, and R. Lam, "Item Recommendation Using Collaborative Filtering in Mobile Social Games: A Case Study," in *IEEE International Conference on Big Data and Cloud Computing*, IEEE, 2014, pp. 293–297.

[96] *The basics of indicator variables*, (Last accessed: 2020-01-12). [Online]. Available: \url{https://online.stat.psu.edu/stat462/node/161}.

[97] F. Thung, D. Lo, and L. Jiang, "Automatic Defect Categorization," in *19th Working Conference on Reverse Engineering*, IEEE, 2012, pp. 205–214.

[98] Y. Tian, M. Nagappan, D. Lo, and A. E. Hassan, "What are the characteristics of high-rated apps? A case study on free Android Applications," in *International Conference on Software Maintenance and Evolution (ICSME)*, IEEE, 2015, pp. 301–310.

[99] *Video: How replays helped Poly Bridge succeed via social media*, (Last accessed: 2020-08-10), Oct. 2017. [Online]. Available: https://www.gamasutra.com/view/news/308236/Video_How_replays_helped_Poly_Bridge_succeed_via_social_media.php.

[100] Q. N. Vu and C.-P. Bezemer, "An Empirical Study of the Characteristics of Popular Game Jams and Their High-ranking Submissions on itch.io," in *International Conference on the Foundations of Digital Games (FDG)*, 2020, pp. 1–11.

[101] T. Wijman, *The World's 2.7 Billion Gamers Will Spend $159.3 Billion on Games in 2020; The Market Will Surpass $200 Billion by 2023*, (Last accessed: 2020-08-12), May 2020. [Online]. Available: https://newzoo.com/insights/articles/newzoo-games-market-numbers-revenues-and-audience-2020-2023/.

[102] F. Wilcoxon, "Individual Comparisons by Ranking Methods," in *Breakthroughs in statistics*, Springer, 1992, pp. 196–202.

[103] Y. Wu, S. Wang, C.-P. Bezemer, and K. Inoue, "How do developers utilize source code from Stack Overflow?" *Empirical Software Engineering*, vol. 24, no. 2, pp. 637–673, 2019.

[104] H.-C. Yang and Z.-R. Huang, "Mining personality traits from social messages for game recommender systems," *Knowledge-Based Systems*, vol. 165, pp. 157–168, 2019.

[105] X. Zhang and A. M. Smith, "Retrieving Videogame Moments with Natural Language Queries," in *Proceedings of the 14th International Conference on the Foundations of Digital Games*, 2019, pp. 1–7.

[106] X. Zhang, Z. Zhan, M. Holtz, and A. M. Smith, "Crawling, Indexing, and Retrieving Moments in Videogames," in *Proceedings of the 13th International Conference on the Foundations of Digital Games*, 2018, pp. 1–10.

[107] A. Zook and M. O. Riedl, "Game Conceptualization and Development Processes in the Global Game Jam," in *Workshop proceedings of the 8th International Conference on the Foundations of Digital Games*, 2013, pp. 1–5.