A Preliminary Study of Game Reviews on The Stem Platform

Abderrahim Mayaba
Concordia University

40114772

Montreal,Canada

ammayaba@gmail.com

Acknowledgements— This paper is a replicate study of the paper "An empirical study of game reviews on the Steam Platform" written by "Dayi Lin, Cor-Paul Bezemer , Ying Zou, and Ahmed E. Hassan", with some modifications of the flow and sections of the study.

Abstract— Computer gaming is one of the most popular industry trends which led to the creation of many large companies. However, this field can be challenging because, as prior studies show, players are extremely difficult to satisfy. Therefore, studying game reviews can help game developers design the games from the user perspective view. Most, if not all, online games platforms give users the option to rate and review the games they are playing. These reviews, provided by users, are usually used by developers as a main resource to discover, and therefore improve, some important defects, things that are users not satisfied with, or maybe even serious bugs that can lead to a system crash. In this paper, I performed an empirical study of a random sample of 5,000 games on the steam platform, one of the most popular gaming platform in the gaming industry, to better understand and help game developers by showing some characteristics that are significant and require more focus than others, and to give some insights that game developers can take advantage of to make the next game development more productive and successful. In this study, I conducted a preliminary study to reveal insights of some characteristic of games reviews, such as the number of game reviews, length of game reviews, and complexity to read through them. The dataset for this study is gathered after crawling the Steam platform official website.

Keywords— Computer games, Steam, User reviews, gaming platforms, Steam Spy.

I. Introduction

The gaming industry is very popular and growing fast which led to the rise of many large organizations. One of the main reasons for the popularity of games are online gaming platform, such as Steam which distribute games online which made games very easy to access.

However, game development can be a challenge, because of the scale of the gaming industry, and as researchers found that computer gamers are people that are usually very hard to satisfy [1]. Some feedbacks from games users can be rich and useful resources to game developers. Most, if not all, gaming platforms give the user the choice to provide a review to the game they are playing which many users take advantage of to post feedback about aspects that they like or don't like about the game. These games reviews can, and usually, contain many significant details like a missing feature, glitch, bug, or even a system crash. When it comes to games, the developer should not only focused on bugs, but also game design which is significantly important to the game users, maybe even more than bugs. It popular among users that games with many reviews are usually popular and has more players that other games. It, also, acts as a score and shows the audience what to expect in the game.

To get some statistics, that can provide some insights and useful tips and help game developers more understand players, and game reviews and how to take the most advantage out of them, I did a preliminary study of reviews on a sample of 5,000 games published on Steam gaming platform. The sample is chosen randomly after programming a customized crawler, written in Python, which crawled the Steam official website and community hub and gathered a large dataset that can be even used for purposes behind the study. I, also, wrote another code, in Python, that queries Stem Spy [2], which is a third-party project that continuously monitor the steam platform, with the chosen 5000 games to get some statistics about them.

Later in this paper, I performed a preliminary study on the number of reviews received each day, the length of the reviews, and the readability of the reviews.

Paper Organization The rest of the paper is organized as follows. Section 2 presents background of the resources used in this study. Section 3 presents and explains the methodology that I used during the study. Section 4 presents the results of the preliminary study, and, finally, section 5 concludes the paper.

II. Background

This section describes in high level the main resources (i.e. websites, libraries, and frameworks) I used to carry on the study.

A. Steam Platform

Steam is a digital distribution, digital rights management, online multiplayer, and communication platform developed by Valve. They are used to distribute games and related media over the Internet, from independent developers to major game companies. In October 2012 Valve expanded the service to include non-gaming software [3]. Steam provides users with the ability to automatically install and manage software across multiple computers, in addition to social features such as friend and group lists, as well as cloud storage and voice and text chat functionality in games. An application programming interface called "Steamworks" is provided that enables developers to take advantage of it to integrate many Steam functions within their software products, including copy protection, network connectivity, achievements in games, and support for content created by users.

Although Steam was initially developed for use on the Microsoft Windows operating system, it has been expanded to include Mac OSX and Linux versions, with limited functionality on PlayStation 3 and both Android and iOS (Apple). In addition to being a central hub for gaming software, Valve has created a version of Steam with modified functions for use in schools for the purpose of educational software, including a modified version of Portal2 for teaching science and critical thinking.

B. Steam Spy

Steam Spy is a website created by Sergey Galyonkin and launched in April 2015. The site uses an application programming interface (API) to the Steam software distribution service that is owned by Valve to estimate the number of sales of software titles offered on the service.

Steam Spy has a straight forward easy-to-use API that can be leveraged to get some statistics about the game, such as the number of players (estimated by range), the current price, initial price, and whether there is a discount at the moment. Steam Spy provides this information by randomly crawls a representative sample of user profile pages.

C. Steam Community [4]

One of the most prominent advantages of the steam platform is the community hub, because it became easy to communicate with other players and create formal or informal groups. Plus, the opportunity for players to create their own personal profile which displays personal details about the player, game details and the number of hours that the player played. It also give players the ability to talk through the microphone with friends, and the player can, also, see her evaluation, which depends on the calculation of the player’s activity in games and the number of hours the player have played, and with the development of this feature it has become easy for the player to obtain a personal signature, and through the signature all users can view the details of the profile and the games the player recently played, as well as the date the player registered on the site, the main groups the player belong to and the country of origin.

The steam community, also, displays all the reviews associated with each game in one page (that you can scroll down to get more reviews until the end of the reviews). It, also, displays, alongside the reviews, some other useful information, such as the time (in hours) the player played the game, the date of the review, and how many games the player has in her own profile.

D. Python [5]

Python is a high-level, easy-to-learn, open-source, extensible, object-oriented programming language (OOP). Python is an interpreted, versatile language, and is widely used in many fields, such as building independent programs using graphical interfaces, and in web applications, and it can be used as a scripting language to control the performance of many programs such as Blender. In general, Python can be used to create simple programs for beginners, but also to create massive projects at the same time. It is often recommended for beginners in the field of programming to learn this language because it is among the fastest-learning programming languages.

Python definitely outperforms other open source commercial programming languages and tools widely used for data analysis, interactive computing and data visualization, such as MATLAB, SAS, Stata, etc., due to its ease and availability of a huge library system to support data science (pandas, numpy, matplotlib, scikit- learn,… etc), besides Python's overall strength in general-purpose software engineering, it is an excellent choice as a primary language for building data science applications.

E. Scrapy [6]

Scrapy is an open-source framework, developed in Python, that is widely used for the development of crawlers. Scrapy has a strong community, offering many additional modules. The first stable version was published in September 2009. Since then, the development team regularly publishes new versions in order to enrich the framework in functionality. The framework has an active community, and commercial support is provided by several companies.

The framework is compatible with Python 2.7 and Python 3.6 or above, on most platforms.

III. METHDOLOGY

This section introduces the methodology of my empirical study of game reviews. I detail how I extracted and processed the collected dataset. Table 1 presents the description of my collected dataset. Figure 1 gives an overview of the methodology I followed.

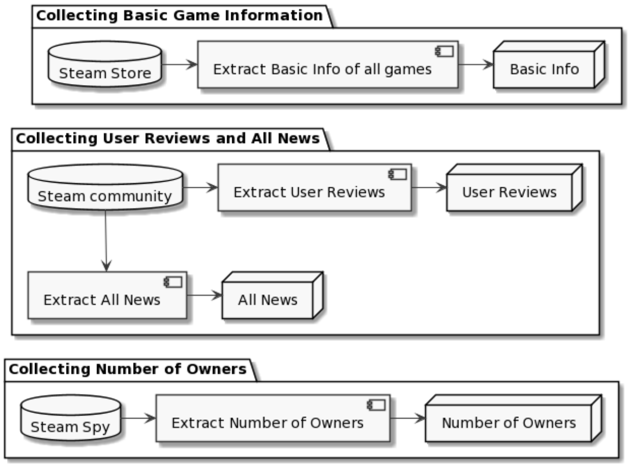


Fig. 1. Overview of The Methodology

A. Collecting Basic Game Information

After deep understanding of the structure of the Steam official website, I developed a customized crawler to crawl all the games published in Steam platform, which are more than 83,000 games. I took advantage of the web crawling framework, Scrapy, which is fast and powerful scrapper. I then ran the crawler on the Steam official website for more than 12 hours and got as much information from every game’s page as possible. I faced some obstacles, such as the inconsistency of Xpath and CSS components which I managed by using Xpath conditions. Also, I faced another problem which is that Steam forward some request to games that contain mature contents to the age checker page. I overcame this problem by using an automation framework named Selenium [7], which has a web driver that you can use to automate some activities like clicking a button on a web page. I programmed the crawler in a way that whenever Steam website forwarded a request to the age checker page, the code, which uses Selenium, would interact by opening the forwarded URL to the age checker page, change the year (to be more than 18), then click the "View page" button, and, finally, hands the control back to the crawler to crawl the page. I excluded games with less than 25 reviews, to avoid a possible bias in the results. I designed the crawler to collect the following information for each game: App id, Title, developer, publisher, tags, genres, number of total reviews, number of positive and negative reviews, number of English reviews, and whether the game is an early access game. The number of games I collected after running the crawler is 24,838 games. After that, I wrote another script, in Python, that takes all the App ids of the collected games, randomly pick a sample of 5,000 games, and then create, both, the associated Steam community URL and Steam community all news URL, which will be the input of another customized crawler to crawl the Steam community to collect all English reviews and all news of the selected games.

B. Collecting User Reviews and All News

To get all English reviews of the selected games, I developed a crawler that takes a list of URLS created from a script I wrote that randomly choose a sample of 5000 games and create the associated community reviews and all news URLS. The crawler start from the review page, that is already filtered to shows only English reviews, and collect all the reviews posted in the page, get the URL of the next page, then crawl until there is no more reviews.

I designed the crawler to collect the following information from every game review item: the body of the review (the review text), whether the review is positive or negative (recommended or not recommend), the number of games the reviewer owns in her profile, the hours the user played the game, and, finally, the date of the review. Because the Steam platform posts the number of playing hours at the present (and not at the time of the posted review), I added another column to the dataset that store the time duration in days between the date of the posted review and the current date. It does that by subtracting the date of today from the date of the game review (I made it dynamic, and not hard coded, so it will give accurate results at any given time). Also, even though the review are supposed to be in English, some of the review are not English, so I had to filter these reviews, using a library named nltk [8], and return only English letters out of them. I, also, removed reviews with no letters but only emojis from the dataset. And, also, removed any reviews with less than 5 letters because, after manually checking some reviews, I found that normally less than 5 to 7 letters reviews do not contain useful feedback but encouragement, such as "Wow", "fun", or the opposite, such as "Bad", which are already illustrated by the rate (i.e. "recommended", and "not recommended") and, as you can see, does not contain useful feedback. The final length of the dataset, after applying all the aforementioned filters, is 574,108 game reviews (with the associated information I mentioned before).

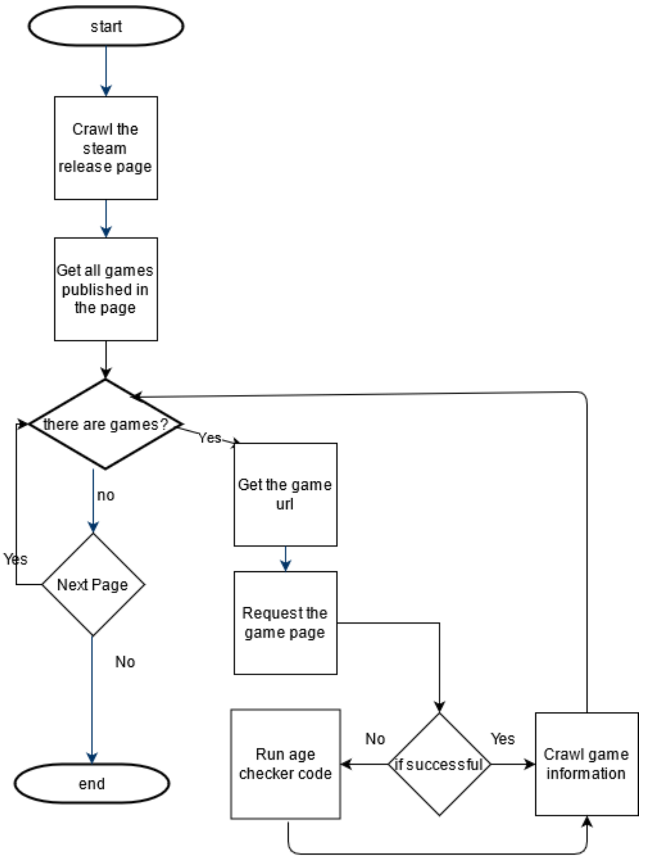


Fig. 2. Flow chart of the games crawler

Element	*Length*
All games (more than 25 reviews)	24,838
Studided games	5,000
All news	32,352
English reviews	574,108
English positive revivwes	48,6914
English negative reviews	87,194
Reviews with approximately accurate playing hours (two days or less)	5,366

Table 1 Dataset Discription

I, also, developed a customized crawler that takes a file that contains URLs of the selected games, the file is the output of the script I mentioned earlier, and then crawl the all-news page associated with that game (URL) to get all the news items associated with the game. I designed the crawler to ignore (and note store) any note that its title does not contain one the words: update, release, patch, hotfix, change log, version, and release notes, to only collect the notes that are related to maintenance or evolution activates. The length of the collected all-news of the selected games dataset is 32,352. I customized the crawler to collect the following information: the news title, the news body, the rate (how many users clicked the "like" button), and, finally, how many users commented in the news.

C. Collecting Number of Owners

To collect the number of owners of all the selected games in my study, I wrote a Python script that queries the Steam Spy API, which is straight forward and easy-to-use API, with the App id and get all the information associated with the queried game. The owners of a game are users who purchase the game and then activate it on Steam, or, alternatively, receive the game through a promo code or as a gift [9]. Please note that the number of owners does not necessarily equal the number of players. The script collect the following information from Steam Spy API: the range of owners, in form of min … max (e.g. 50,000 ... 100,000), the current price, initial price, and the percentage of the discount, if there is any.

D. Types of Studied Reviews

To distinguish between different types of games, I made use of the genres that are provided by the developers. I considered games with “Indie” genre as indie games, and games with “Free-to-play” genre as free-to-play games. I, also, noticed that when the game is in early-access mode, the page of the game has an extra html component that is used to display the early access details, so I used that as an indicator of whether the game is an early-access-game. In my study, I compared all the studied reviews along the four dimensions:

1. Positive reviews and negative reviews

I studied the length of positive and negative reviews, and whether they are different from each other

2. Indie-game reviews and non-indie game reviews

Indie games in a significant part in gaming platforms, as they provide a way to manage developing games from small studies with small budgets. I classified indie games using the genres tags, which is provided by the developer (and not “tags” which is provided by players). I studied whether indie-games and non-indie games are different from each other.

3. Early access reviews and non-early access reviews

In early access games, the developer gives players the option to purchase and play a game in the various pre-release development cycles [10]. I studied if early access reviews are different from non-early access reviews.

4. Free-to-play game reviews and non-free-to-play game reviews

I studied whether paying for a game would impact the user reviews on the game.

I analyzed and compared the reviews for each of the aforementioned categories.

IV. Preliminary Study of The Characteristics of Game Reviews

Understanding the characteristics of game reviews are an important tasks because it will help the game developer assign, efficiently, the resources to go through the top significant game reviews, instead of manually reading all the reviews which is a hard tasks especially if the game is popular and purchased by many players. Game developers should pay attention to game reviews, at any given day, and get as much useful information as possible from them because game reviews are one of the primary forms of videogame journalism and are, also, one of the prevalent forms of discourse about games. Again, this task can be done efficiently when focusing more on important aspects.

In the preliminary study, I studied the number of reviews that games revive each day, the length of the reviews, and the readability of the reviews. After that, I investigated the effect of different game characteristics on the daily number of reviews.

A. How many reviews are posted and what is their complexity?

Approach I analyzed the complexity of game reviews from three perspectives: the number of reviews to read each day, the length of the reviews, and the readability of the reviews. To compare the results of the different types of game reviews, I calculated the median of all categories and compare them. The game reviews are grouped as follows: positive vs negative reviews, early access games vs non-early access games, free-to-play games vs non-free-to-play games and indie games vs non-indie games.

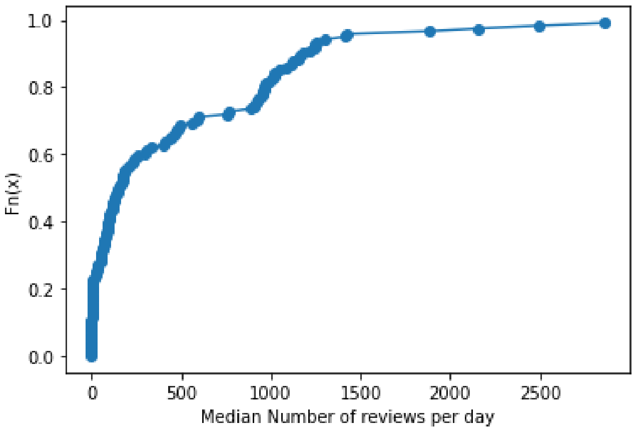
I, also, measured the readability of reviews using the Coleman-Liau index. the Coleman Liau Index is designed to evaluate the U.S. grade level necessary to understand the input text. The score you receive indicates the U.S. school level a person needs to understand the text. So if you receive a rating of 9.5, your text should be easily understood by a student in 9th or 10th grade [11]. Therefore, the higher CLI index means that the text is harder to understand, and vice versa. Hence, game reviews with low CLI index are easier to read and understand. Table 2 shows an example of game reviews with a high and low CLI index, from the dataset, respectively.

Review content	***CLI***
"I adore already and this is a prime example of why . this is going to make many amazing and I am so happy for him . Please make more and I will donate to your if you do . I this game and if you ' re reading this it ..... its worth it."	2.14
“Democratic Socialism Simulator is a wonderfully approachable political game that , perhaps optimistically , its player to explore Socialist and Progressive and their . It is an absolutely cathartic balm .”	18.67

Table 2 Example of reviews with high and low CLI

Findings Around 80% of the games receives a median of just below 1,000 game reviews per day. Figure 3 displays the cumulative distribution function of the median number of the daily game reviews.

As you can see, the number of reviews received every day is very high, which makes the process of reading every review very difficult and resource consuming.



Most games receive game reviews with a median length of 104 characters. I calculated the text length of each game review after removing reviews with less than 5 characters, because, after some manual checking, usually these reviews are not useful to developers.

Negative game reviews are longer than the ones of positive reviews with a median length of 218 and 112 characters, respectively. As illustrated by the median numbers, the difference is significant.

Indie games have a slightly higher median of the length of reviews than non-indie games with 344 and 303 characters, respectively.

Early access games receive shorter game reviews than non-early access games with 296 and 344 characters, respectively. As you can see, the difference is not that much, and it is approximately the same difference as between indie and non-indie games.

Not surprisingly, players write longer game reviews for paid games than free-to-play games. The medians for free-to-play and paid games are 188 and 356 characters, respectively.

All game reviews have a median readability (CLI index) level of grade 4. The score is quite low, which means that the reviews are easy to read and understand.

V. Conclusion

Developing softwares with high quality is usually the ultimate goal for any software engineer. However, when it comes to gaming development, the customers (i.e. players) are very hard to please and they would focus on more aspects than the quality of the game, such as the design of the game and the content itself.

In this paper, I performed a preliminary study on a selected games from Steam platform, after crawling the official website and the community website using a programmed web crawler, designed especially for this study. The results can be useful for game developers and, also, give the developer a better understanding of how to interpret the reviews efficiently using available resources. This study, also, gave me the opportunity of understanding more how to get the dataset, and gave me the chance to develop a customized web crawler, that can handle some unexpected responses like age checker pages, for the first time. It, also, allows to learn to better automate extracting dataset and handling them.

References

[1]. Chambers C, Feng Wc, Sahu S, Saha D (2005) Measurement-based characterization of a collection of onlinegames. In: Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement, USENIX Association, pp 1–1

[2]. www.steamspy.com

[3]. https://en.wikipedia.org/wiki/Steam_(service)

[4]. https://steamcommunity.com/

[5]. https://www.python.org/

[6]. https://scrapy.org/

[7] https://www.selenium.dev/

[8] https://www.nltk.org/

[9] Sergey G (2016) SteamSpy - All the data and stats about Steam games. http://steamspy.com/, (last visited: Mar 29, 2018)

[10] https://en.wikipedia.org/wiki/Early_access

[11].www.webfx.com