Final Group Report

CS489 Computer Ethics and Social Issues – Team 4

20170180 Chulhwan Kim, 20200212 Suhyeon Ryu, 20200633 Haeseul Cha

Abstract

In our final group report, we address the pervasive issue of algorithmic bias and its implications for discrimination in various domains such as education, selection, and media. Focusing on the risks associated with biased algorithms, including the reinforcement of prejudice, discrimination, and potential abuse, our project aims to raise public awareness from both platform developers' and content consumers' perspectives. To achieve this, we developed an interactive game using React and TypeScript, where players assume the role of a recommendation algorithm within a media platform. The game emphasizes the impact of content consumption on consumers and the company, incorporating elements of polarization and social conflicts. Technical details include the use of 8-bit retro design assets, drag-and-drop functionality, typing animation, and various libraries such as React DnD, react-typist, Router, and Recoil for state management. The game structure involves rounds and waves, each presenting challenges related to content recommendation and consumer reactions. Consumer preferences, content topics, and events contribute to the complexity and reality of the game. The project concludes with a discussion of identified issues, such as the need for improved visualization of bias and enhanced gamification elements, and outlines future plans for development and refinement.

Here is our GitHub repository: https://github.com/Cathy-CHS/Feed_Lovers

Keywords: Algorithmic Bias, Discrimination, Polarization, Content, Consumer Preferences, Gamification, Retro Game, Pixel Art.

1. Problem Definition

Algorithmic bias is currently leading to discrimination in various areas such as education, selection, and media. This issue arises from training with biased data or using only a portion of the training dataset. Examples include sanctions on adjusting search algorithms by Naver and the hate speech response of Microsoft's chatbot service 'Tay' towards societal vulnerabilities. Algorithmic bias poses dangers such as unquestioning acceptance of bias by consumers, reinforcement of prejudice and discrimination, the potential for systematic AI abuse, and declining reliability in applying algorithms to critical systems like employment and healthcare.

Recognizing the risks, our objective is to raise public awareness about algorithmic bias from two perspectives: first, questioning the morality of recommending content for profit using algorithms from the standpoint of platform developers, and second, urging content consumers to evaluate algorithmic recommendations critically.

2. Method

To achieve our objective, we developed a game to represent the concerns in the form of a game, involving an interactive experience. Our focus was on portraying the impact of content consumption on consumers and the company by making game players be the algorithm ‘itself’ of the corporation to earn money, emphasizing the polarization of values through indiscriminate content consumption. We developed the game using React and TypeScript, consisting of 5 pages, 19 components, and 23 images. Also, additional libraries such as recoil, typing animation, and router are actively used to make the concept of the game more seamless.

The game’s development process involved conceptual planning, detailed refinement of scripts and phases, and implementation. First, we developed an overall layout in the concept of a media platform, which contains a list of consumers and contents. Second, we categorized content that people enjoy as five themes. Then, we researched various issues and articles that are related to each content. These social topics are intended to draw the attention of content consumers and trigger conflicts with them. We decided to contain this as a form of news script in the game.

Game pages are constructed as guide, main, and ending. We divided rounds and waves, and then developed scripts for each level. After setting big frames, we adjusted the difficulty and balance of the game by controlling the timer and the goal of invested money.

3. User Game Scenario

The game commences from the lobby page, progressing through a guide, main, and ending page. Starting at the guide page, users are engaged through a command prompt conversation with an unidentified entity, revealing that they are assigned the role of a recommendation algorithm.

Users move to the main page to start the game. Following an explanation of a sudden command, users are tasked with 'recommending' contents in the right box to 16 consumers. As five waves unfold, the contents are continually updated, with most consumers having preferences, while a few receive all types of content. As a recommendation algorithm, users aim to earn money by skillfully dragging and dropping suitable content to each consumer. However, both algorithms and consumers can be influenced by news or gossip. Introducing a news event limits the available content, and consumers exhibit varied reactions to their preferences with the same content. Some consumers, having encountered incorrect recommendations or hate speech, opt to cease using the service. Successfully passing one round requires users to repay all investments before the conclusion of each wave.

At Round E, which is the last round of the game, the company ‘Feed Lovers’ and game users are regulated. Users encounter the ending scene after the round E ends. The game screen shows the endless conflicts of users, and the command prompt suggests the game's message, asking for our opinions on the algorithms we have been doing so far.

4. Technical Details

- **Assets**

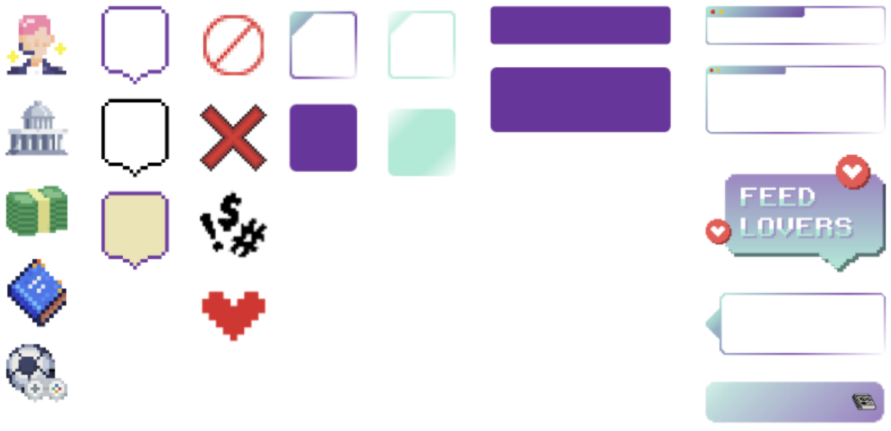
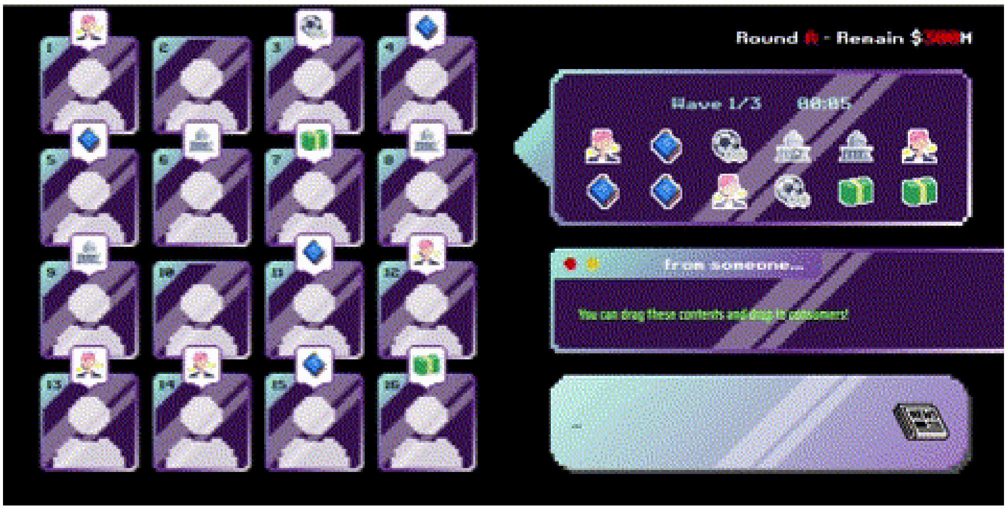


Figure 1. Image assets

The image assets for this project encompass various elements, including icons representing different content types, the grid and background for each textbox, our logo, and additional components. Since the overall design concept of the game was set to 8-bit retro, assets were also designed accordingly. The creation of these assets posed a challenge, particularly in achieving a pixelated design, leading to the need for manual revisions on some occasions. Special care was taken during the development of content icons. To achieve the desired pixelation effect, we developed and used a program that not only pixelates an image but also adds a white outline to the pixelated version. Furthermore, each icon underwent manual modifications to enhance visibility and ensure a cohesive visual presentation.

- **Drag & Drop**



You can drag these contents and drop to consumers!

Figure 2. Drag & Drop Interaction

The pivotal technical aspect of the game was the implementation of drag-and-drop functionality. This choice was made to intuitively depict the process of feeding content. This feature played a crucial role in almost every interaction between the game and the user, prompting us to prioritize creating an engaging and dynamic drag-and-drop experience. We employed React DnD for this purpose.

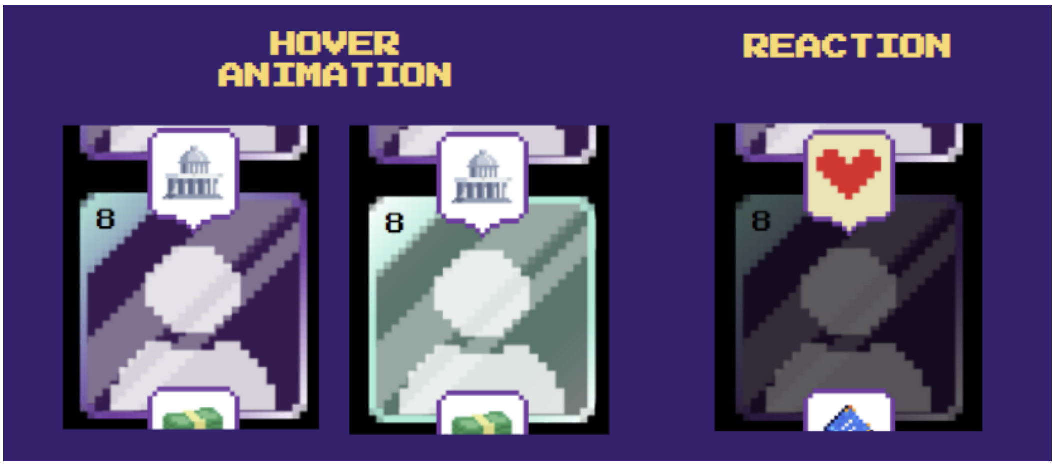


Figure 3. Different visualization by hovering and reaction

To achieve an animated and lively drag-and-drop experience, we employed two techniques. The first technique involves hover animation, creating visual feedback when the user hovers over an icon. The second technique incorporates reactions from consumers. We designed two distinct sets of consumer grids, dynamically switching between them based on whether the icon was being hovered over. Additionally, consumers displayed reactions when content was dropped onto them, achieved through the manipulation of various states and the use of ternary operators. This approach aimed to enhance user engagement and create a more immersive gaming experience.

- **Typing Animation**

For a more realistic production according to the retro game theme, a typing animation library 'react-typist' was applied to the command prompt text.

- **Router**

To reuse components, we imported the Router library in React. It enables the creation of flexible and reusable components that can respond to changing URL parameters, allowing us to build dynamic and responsive user interfaces.

- **Recoil**

Recoil is a state management library for React applications. It provides a simple and efficient way to manage the state of React components. It aims to address some of the challenges associated with state management in large and complex applications. We set key numerics such as rounds, contents, events, and consumers' preferences into atoms, which are units of state. Then, we applied those atoms into components and pages in different files to enable managing states globally. It allows the game to immediately reflect real-time updates that occur in the game interactions.

5. Structure & Deliverable Message

a. Intro Animation

The intention for the intro animation was to evoke the feeling of starting a console game, aiming to provide immersion as players transition to a different world.

b. Lobby Page

The initial plan was to show buttons selectively based on whether users watched the guide, game, or ending. However, the logic for checking this decision has not been implemented. since this decision-making logic was not applicable after deploying the game on the web on a desktop app.

c. Guide Page

Considering that players may feel unfamiliar with 'becoming' the algorithm, we deemed it necessary to provide additional explanation. The terminal window and command prompt screen messages are associated with the corporation director, making it easy for players to infer their role. We added typing animations to enhance engagement and make users focus on the text. This is intended to prevent players from missing information while quickly reading the output messages and finally concentrating on their role and each character.

d. Main Page

i. Rounds

We adopted alphabetic indexing (A, B, C) instead of numerical indexing (1, 2, 3) to draw an analogy to startup investment steps. Target repayment amounts were set using a sigmoid function, mirroring the investment amounts in the startup series. Users have to earn profits by using all the money invested through content recommendations. Because we set it to progress forcibly to an ending in Round E, currently there are only four playable rounds, A to D.

To better reflect actual investment stages, we also considered scenarios in which multiple investments occur in one series such as in the form of round B' and D'. We intended to convey situations where the investment amount is low or the repayment period is urgent, depending on the investment environment, along with news articles. However, we passed this on to the next development process.

ii. Waves

We applied five waves for each round to prevent monotony in the playing pace. Each wave introduces a time limit and new content. Players determine the number of contents to recommend for each wave by dividing the target amount by the number of waves, thus learning how many contents to recommend for each wave.

The difficulty is adjusted based on the wave's time limit and the round's target amount. The time limit for each wave in one round was set to be the same. The content list consists of 2 rows with 6 columns each.

- Early Stages (Round A~C): Due to the progressively increasing repayment amount, the time limit also increases linearly. However, as the user starts moving the content list from the inside, the time required increases quadratically with the number of contents. Therefore, the perceived difficulty by the user increases more steeply than a linear graph.

- Later Stages (Round C~E): While the investment amount hardly increases, only the time limit decreases. Users, now familiar with drag-and-drop matching games, will smoothly progress even with reduced time. For the special interaction in C, a single drop yields 16 times of the normal profit. Matching 1 row with 2 items takes 2 seconds, increasing by 1 second for each subsequent row. The time required to move all 12 contents given in one wave ranges between 20 and 25 seconds. For Round D, which requires the highest difficulty, we adjusted the difficulty by setting a time limit of 20 seconds per wave to move 10 contents per wave for 3 waves.

iii. Contents

There are five broadly divided content topics that are likely to be of interest and accompanied by various issues in society: Sports and Games, Entertainment, Self-development and Tips, Economy and Stock, Politics and Society. Importantly, the chosen social topics are strategically designed to capture the interest of content consumers and stimulate conflicts or discussions among them. These thematic selections aim to not only entertain but also provoke engagement and reflection, particularly in the context of algorithmic bias and its societal implications.

iv. Consumer Preference

Every consumer has a preference for each content, represented by filling their character image with a specific color. The major color of each content icon corresponds to the color of their preference, allowing players to intuitively grasp each consumer's inclination. The default white color indicates neutral preference. Consumers with high neutral preferences tend to consume various content rather than focusing on specific ones.

The total sum of each consumer's preferences is 100, with the initial values determined as follows:

- Neutral preference is randomly assigned between 0 and 100.

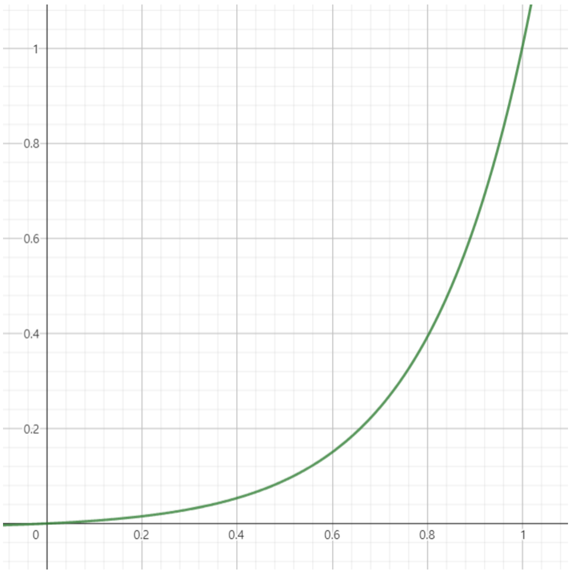


Figure 4. Different visualization by hovering and reaction

- The remaining portion is filled with content-wise preferences, each proportional to its weight. Weights are determined by an exponential probability distribution function (pdf).

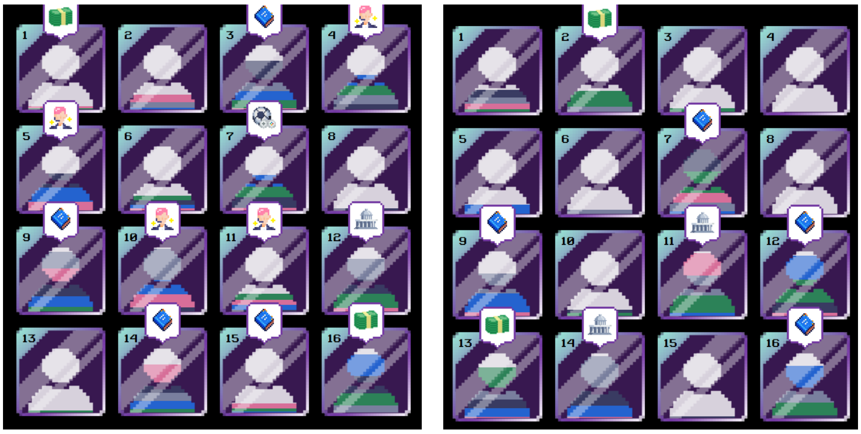


Figure 5. Different ratios of content-wise preferences by pdf (linear: left, exponential: right)

- Using exponential pdf results in greater differences between weights, allowing for the representation of diverse consumer preferences. A comparison between linear and exponential pdfs shows noticeable differences.

- Neutral preferences are placed at the top, followed by content-wise preferences in order of their respective weights from highest to lowest.

In regular waves without events, consumers may demand specific content based on their preferences.

- Neutral preference below 60: Consumers with lower neutral preferences demand a particular content they are keenly interested in. The probability of demanding each content is proportional to their preference.

- Neutral preference 60 or above: Consumers with higher neutral preferences may not demand specific content. The probability of not demanding content is related to their neutral preference, while the probability of demanding content is proportional to the corresponding preference.

When consumers acquire content, their reaction affects their preferences.

- Satisfactory content acquisition: Preference for the acquired content increases by 20, while neutral preference reduces by 20. If the neutral preference is initially below 20, it becomes 0, and the remaining reduction applies proportionally to the preferences of other contents.
- Unsatisfactory content acquisition: preference for the acquired content becomes half. The neutral preference increases as the content-wise preference decreases.

v. Events

Events occur in the first wave of rounds C and E. The news event leads to the store being filled exclusively with content related to the topic. In this scenario, consumers demonstrate a heightened sensitivity to current issues and gossip. While the event happens, the command prompt aggressively pushes players with a context that suggests, ‘Even if consumers leave due to conflicts, as long as we make money, that's all that matters.’. This reflects the attitude of companies that focus solely on immediate profits using algorithmic bias.

The event in the C-1 wave primarily conveys the theme. Consumers who have encountered news about game addicts all desire game-related content. We obviously cannot expect positive reactions from users to this gossip. Initially, the intention was for consumers who usually prefer game content to be criticized by other consumers, leading to their departure. However, as this preference wasn't visually apparent, the logic was slightly adjusted. When exposed to provocative news, satisfaction varies among consumers, with only one person being content and happy with the specialized content, while others express dissatisfaction and deliver negative remarks. As a result, some users leave the platform during the argument. This illustrates the user displacement caused by conflicts, contributing to a decline in service effectiveness.

The event in the E-1 wave connects to the ending scene. It starts with news that the player's platform is being regulated.

More events were planned for a more immersive experience, such as reduced investments due to the economic downturn in the D-1 wave. However, it has been postponed to future development due to other priorities.

e. Ending Page

The ending messages convey the fall of the platform by the hate speech regulation. Reflecting on the journey, the platform questions why it failed despite meeting consumer demands. Despite the failure, the platform acknowledges that people will still seek its content in the algorithmic era. The closing message gives a hint of the possibility of a new platform with the same actions, creating an implication when users start a new game.

6. Discussion & Future Plan

One significant issue identified in the current system is the lack of visualization, particularly in understanding the bias or loyalty of consumers toward the virtual service. The confusion arises as to why individuals resort to blaming and cursing each other when receiving desired content, and why some continue to enjoy such content despite the negative reactions around them. This lack of clarity becomes more significant during news events, leading to a challenge in comprehending player responses. We tried to solve this issue by applying the consumer preference feature. However, thinking about ways of more striking visualization should be considered.

The existing gamification strategy relies solely on displaying the remaining money to survive, which may not be sufficient to motivate users effectively. This limitation poses a challenge in sustaining user engagement and interest in the game. To enhance the gamification aspect, we conceive more UI to support the recognition of the goal. One example is visualizing the total amount users had earned. With this numeric, players can be encouraged to aim for higher achievements, fostering a desire for more challenging gameplay. This modification aims to boost user motivation and maintain their interest throughout the gaming experience.