The Rhetoricon Project: Creation of GoFigure

By: Evan Hum (STU#20841899)

Executive Summary

The Rhetoricon is a research group focused around studying the complexities of language, namely rhetorical figures, and its impact it can have on the human brain. While the Rhetoricon Project contains many different aspects and components, this technical brief will focus on GoFigure. GoFigure is a citizen science-based game focused on gathering real data about rhetorical figures through gameplay. The main flow of the game involves a player finding a piece of text that they feel has some sort of linguistic pattern to it, then going through the process of submitting their instance for a chance to receive points upon approval. If their submission is approved, it gets sent to the Rhetoricon database where it can be used for research purposes.

Regarding GoFigure, this brief will cover 3 sections including the design, the building, and the future of GoFigure. The design section focuses on the needs of the project and brainstorming what would be required to support the addition of a game to the existing database. The building section covers the different tools that were used to develop the game along with explanations as to how the design ideas were implemented. Finally, the future section describes the next steps for the GoFigure game including new features that will be developed. It outlines the planning that went into supporting these features and the reasoning as to why they should be added.

Introduction

The Rhetoricon research group was created to explore the ways in which rhetorical figures are used in all types of literature and languages to develop theories and truths about the way the human mind works. “Rhetorical figures [are] patterns that are present in all languages, reflecting both neurocognitive affinities and communicative imperatives, are linguistic devices that serve mnemonic and aesthetic purposes, alongside informational and argumentational purposes [1]”.

The Rhetoricon Project consists of 2 main components. The Rhetoricon website contains information regarding the types of rhetorical figures and how they can interact with each other to form new figures or the ways in which they can be related through linguistic domains and cognitive affinities. The GoFigure website is a citizen science-based game revolved around inviting users to investigate everyday text they see, looking for possible patterns and then submitting them as instances to be annotated and added to the Rhetoricon database.

This technical brief will focus on the GoFigure game including an overview of how the game works in its entirety, the history of the initial version of the game and what went wrong with it, the planning that went into the redesign of the game, and how it aims to improve on the previous version. The process of building a new version of the game along with technical details and technologies used, and finally, the plans for the future of GoFigure.

GoFigure Overview

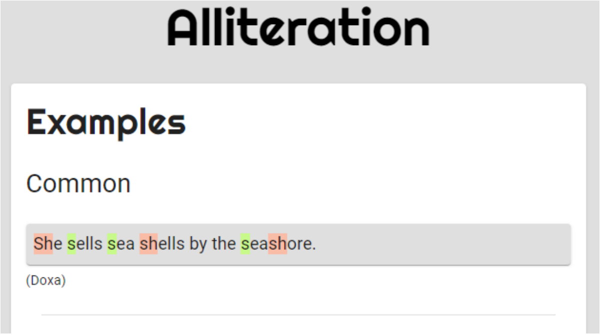
The GoFigure game is a citizen science-based game focused on gathering real data about rhetorical figures through gameplay. The core functions of the game include scouting an instance, marking the instance, and getting a moderator to review the instance. Other important features are tutorial videos for introducing players to the game, a field guide where players can look up figures and learn more about them, and a point system where players can see their own points and view who is at the top of the leaderboards.

Scouting

Scouting an instance is the process of examining pieces of text and discovering a pattern within it. This can range from reading a book, article, or report to scrolling through social media or watching a YouTube video. Once a piece of text with good potential for containing a rhetorical figure is chosen, the player will fill out the scouting form including citation information and a genre selection. The instance is then submitted and is now ready to be marked.

Marking

Marking an instance is the process of selecting an appropriate rhetorical figure and highlighting the portion of text that exemplifies that figure. Based on the type of figure the number of highlights and colour options will vary to reflect the intricacies of the figure. A sample of a marked instance can be seen in Figure 1. This is known as an annotation.



Alliteration

Examples

Common

She sells sea shells by the seashore.

(Doxa)

Figure 1: Sample of a marked instance taken from GoFigure Field Guide.

Reviewing

Finally, there is the review system. Once an instance has been marked, the instance will be given a status to let the moderators know it is waiting for a review. The moderator will look at the marked instance and determine if the selected figure is correct and that the highlighted sections are in the right places. If both hold true, the instance annotation will be approved and sent to the Rhetoricon database.

Designing the Game

The design of the game was the most important and time-consuming portion of the creation of GoFigure. Due to the nature of having both a frontend and backend, bridging the gap between the two types of development proved to be quite complex. The following sections describe the specifics of what was designed in both the frontend and backend.

The Frontend Design

The frontend involved designing both how to game should look and function and use cases to plan out all the different ways users could possibly interact with the game interface. The UI was designed using a software called Figma, where design templates could be created to visualize possible ideas of what pages should look like (homepage, scouting page, marking page etc.). “[A] use case is a description of all the ways an end-user wants to “use” a system. These “uses” are like requests of the system, and use cases describe what that system does in response to such requests [2]”. For GoFigure, use cases were focused on the different functions of the game and brainstormed all the ways a user could attempt to use the function, and how the function should handle it. For example, the marking use case focused on the process of adding highlights to an instance and what would happen if an incorrect number of highlights were added.

The Backend Design

The backend involved planning out what APIs would be needed to support users submitting data to be stored as well as the creation of new game specific tables. Previously, a database was created using phpMyAdmin to manage tables and store data. To support the GoFigure game, new tables were needed to store information before it could be approved and sent to the main Rhetoricon tables. These tables would hold information like submitted instances and their citation information, and once a moderator approves of that instance, its data can then be copied over to the Rhetoricon tables. The next for being able to use the new tables was to create APIs. An API is the intermediary software that allows two applications to talk to each other [3]. In the case of GoFigure, it allows the frontend to communicate with the tables in the backend. A document was created, detailing all of the APIs that would be needed for the game to function as well as simple descriptions of what information it would need from the frontend (the request) and what information it should give back from the backend (response). An example of an API that would be needed is a registration API. It takes the users email and display name, adds it to the users table and returns the display name, level, and total points on their profile page.

Building the Game

The main technologies used for the GoFigure game consist of Nextjs, Typescript and golang with help from frameworks and packages such as ToastUI, MaterialUI, remark, eslint, gorm and many more. The process of building the game was split into 2 steps.

Backend Building

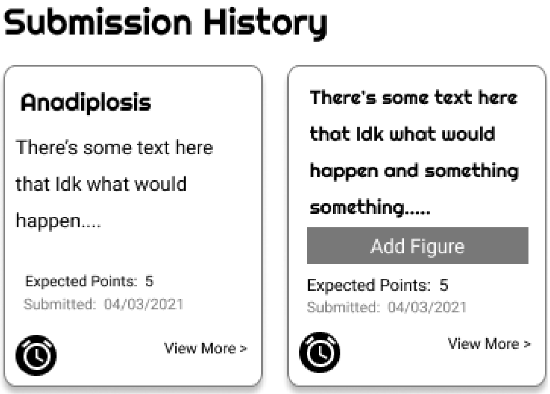
First, there was the backend building. This included the creation of new database tables to support game submissions as well as the building of the APIs to interact with those tables. Some of the tables that were required to support the game included a game_examples_text, game_examples_figure and game_source table for temporarily holding the instance and annotation information while the submission is waiting to be approved. Then once it is approved it will be copied into the Rhetoricon examples_text, examples_figure and source table. Also, a game_points_events table was created for connecting the user’s id to their submission so when they get approved, the points can be awarded to the player. Once the tables were complete, APIs were needed to access the tables and fill them with data. The main APIs to support core functionality consisted of APIs for retrieving an instance from the database, adding an instance to the database, adding an annotation to the database, allowing a moderator to update the status of a submission, and most importantly, APIs for registering users and fetching their information upon login.

Frontend Building

Secondly, there was the frontend building. This included creating interactive pages for users to be able to play the game as well as utilizing the APIs that were built to take the data from the users and store them in the tables. The core features that were built were a login/registration page, a player home page, a scouting page, a marking page, and a moderating page. The login page utilized the user profile API to check if the user existed in the database, if the user did not exist, it called the register API allowing the user to create an account. After logging in, the player is brought to the player home page which uses the response from the API to display the user’s statistics. The next page that was built was the scouting page. This page was designed using MaterialUI’s form component and allowed a user to enter in their instance text and citation information. Once submitted, it would call the instance creation API to add the data to the game tables. The next page is the marking page. This page was exceptionally complex for allowing a user to highlight a certain section of their text and store the correct index values for their highlight. Referring to Figure 1, the first highlight on “She” would be stored at index [0,1] with 0 being the start and 1 being the end. Through the use of built-in typescript functions such as window.getSelection() [4] and recursive functions, an interactive highlighter was made. Once a player finished their marking, the index data was then passed through the create annotation API to add the information to the respective tables. Finally, the last core component was the moderator page. Using MaterialUI’s table component, a list of all pending instances was created which allowed for a moderator to select a specific instance to review. An interface was created using basic buttons to allow the moderator to approve and reject the annotations that belonged to the instance and then calls the status update API to change the status of the instance and its corresponding annotations.

Future of the Game

Currently, the GoFigure game is still under early stages of development. Before it can be released publicly, there are some important features that need to be developed. The first feature involves allowing a player to review their submission history from their profile and add new annotations to previously marked instances. The API for this feature has already been created which retrieves a user’s submissions, but frontend support is still required to display this information on a page. A Figma prototype for this page has been developed and can be seen in Figure 2.



Submission History

Anadiplosis

There’s some text here that Idk what would happen....

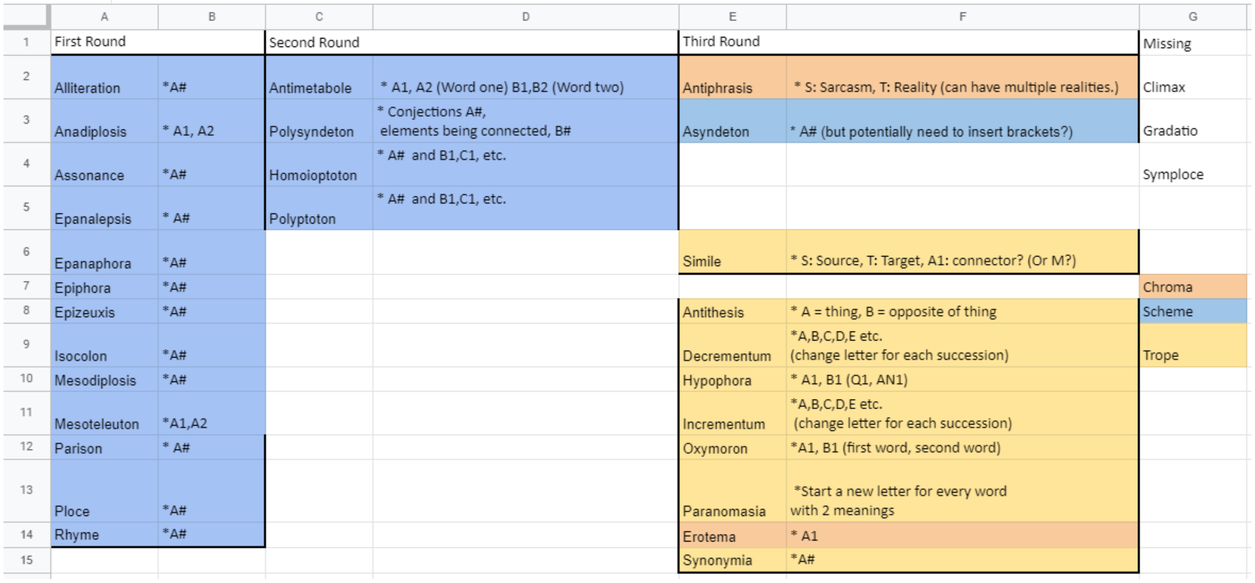
Expected Points: 5
Submitted: 04/03/2021

There’s some text here that Idk what would happen and something something.....

Expected Points: 5
Submitted: 04/03/2021

Figure 2: Figma prototype for submission history on mobile

The next important feature is adding support for a wider variety of figures. Currently, only a handful of figures exist for a user to choose from based on the limitations of the highlighter component and the simplicity of the figures. As the game progresses, more and more figures with complex patterns will be added as users become more familiar with identifying rhetorical figures. A spreadsheet of the order new figures will be added can be seen in Figure 3.



	A	B	C	D	E	F	G
1	First Round		Second Round		Third Round		Missing
2	Alliteration	*A#	Antimetabole	* A1, A2 (Word one) B1,B2 (Word two)	Antiphrasis	* S: Sarcasm, T: Reality (can have multiple realities.)	Climax
3	Anadiplosis	* A1, A2	Polysyndeton	* Conjections A#, elements being connected, B#	Asyndeton	* A# (but potentially need to insert brackets?)	Gradatio
4	Assonance	*A#	Homoioptoton	* A# and B1,C1, etc.			Symploce
5	Epanalepsis	* A#	Polyptoton	* A# and B1,C1, etc.			
6	Epanaphora	*A#			Simile	* S: Source, T: Target, A1: connector? (Or M?)	
7	Epiphora	*A#					Chroma
8	Epizeuxis	*A#			Antithesis	* A = thing, B = opposite of thing	Scheme
9	Isocolon	*A#			Decrementum	*A,B,C,D,E etc. (change letter for each succession)	Trope
10	Mesodiplosis	*A#			Hypophora	* A1, B1 (Q1, AN1)	
11	Mesoteleuton	*A1,A2			Incrementum	*A,B,C,D,E etc. (change letter for each succession)	
12	Parison	* A#			Oxymoron	*A1, B1 (first word, second word)	
13	Ploce	*A#			Paranomasia	*Start a new letter for every word with 2 meanings	
14	Rhyme	*A#			Erotema	* A1	
15					Synonymia	*A#	

Figure 3: Spreadsheet of figure implementation order

Some other features that have been planned out but are not necessary for the first release of the game is a leaderboard to allow players to see their standing compared to the other players, and support for classroom use. This entails allowing a teacher/professor to create a class where players can join the class either through email invitation or a classroom code. The purpose of having a class is so that the teacher/professor can moderate all their students' submissions without having to search through all player submissions. Students can also have their own classroom leaderboard to see how they are doing relative to their peers and be evaluated for a portion of their final grade.

Summary

In summary, the citizen science-based game, GoFigure, is still in the early stages of development. However, after extremely detailed planning around the creation of new database tables accessible through new APIs, the developments have gone smoothly and according to plan. All the core features and supports have been built to allow GoFigure to run as a minimally viable product. Furthermore, key developments for the future have also been planned out and accounted for so that the current state of the game can incorporate these new changes without having the rework any of the existing functions.

References

[1] R. A. Harris and C. Di Marco, "Rhetorical figures, arguments, computation," IOS Press, Waterloo, 2017.

[2] E. Larson and R. Larson, "Use cases: what every project manager should know," in *Global Congress 2004 - North America*, Anaheim, CA, 2004.

[3] MuleSoft, "What is an API? (Application Programming Interface)," 2021 MuleSoft LLC, a Saleforce company, [Online]. Available: https://www.mulesoft.com/resources/api/what-is-an-api. [Accessed 25 April 2021].

[4] MDN Web Docs, "Window.getSelection()," 2005-2021 Mozilla and individual contributors, [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/API/Window/getSelection. [Accessed 29 April 2021].