Tabletop Trove: RAG System for Board Game Recommendation

Kasey La

1 Introduction

The goal of this project is to create a recommendation system using RAG that can help users choose a board game based on their preferences via a query and filters. Typically when looking for board game recommendations, the best option is someone who works at a board game store or a friend who is familiar with many board games due to their domain-specific knowledge. Given a large amount of domain-specific data, can a RAG system provide good board game recommendations?

2 Data

The dataset is the Board Games Dataset from Kaggle[1] which contains various information for 90,400 board games and board game expansions retrieved from BoardGameGeek in 2017. Among the dataset, 76,688 are base board games and 13,712 are board game expansions.

For the cleaning steps, only the relevant columns were extracted: ID, name, description, category, mechanic, minimum and maximum number of supported players, minimum and maximum amount of playtime, minimum age, type (whether it is a base game or an expansion), expansion (the base game upon which an expansion builds), and average rating (1-10 on BoardGameGeek). Across all textual columns, escaped characters were replaced with their respective unicode characters and newline characters were removed. Within just the descriptions, a few had html comments that did not provide contentful information about the game, thus those comments were removed. Within the columns containing lists (i.e., category, mechanic, and expansion), there were many delimiter issues, so those were fixed as well. The columns containing integers (i.e., those that have to do with number of players, amount of playtime, and minimum age) and the average rating column were reinforced to be integers and floats respectively.

The importance of retaining numeric data types for the numeric categories is for the filtration portion of the project, which requires the data to be treated as numbers. On the other hand, since numbers are not very semantically informative on their own, additional columns were created to provide more context, such as “60” becoming “Maximum playtime: 60 minutes”

[1] https://www.kaggle.com/datasets/gabrio/board-games-dataset/data

and “6.5” becoming “Average rated game (6.5/10.0)”. These additional columns were then added to each entry. For many of the categories, some cells did not have a provided value. For the numeric columns, 0 was a possible value. In prose, this does not make any sense since a game cannot possibly have, for example, both a minimum and maximum of 0 minutes of playtime, so all zero values were written as “Unknown” in their contextual columns. There were also some games where the minimum number of supported players is nonzero while the maximum number is zero, which is another issue in itself; however, it is not relevant for the dataset part but rather the language generation part. These issues were dealt with in the filtration process.

3 Model Architecture

The user enters a query into the front-facing web page built using Streamlit. The query is embedded using all-MiniLM-L6-v2, and the top 20 results are retrieved with FAISS. The user then optionally adjusts the filtering options. For these options, there are checkboxes that filter out expansions, include games that support over 12 players, and include games that have playtime over 360 minutes (6 hours), and there are sliders that allow the user to select the desired minimum and maximum number of players, playtime (in minutes), and average rating that they would like the result to have. The remaining results that meet these criteria are then written in prose format and fed into Llama-3.2-1B Instruct (with temperature set to 0.3) to produce a game recommendation for the user.

4 Experimental Setup

The experimental phase involved collecting a list of 20 human-generated queries to feed into the system in order to evaluate both the vector search result as well as the generated answer. Each query was entered into the system along with basic filter options selected and then the recommendation and the selected search result from which the recommendation was based were extracted for evaluation purposes.

Human evaluation was used to judge whether the selected search result is valid for the given query (i.e., Does the selected game actually have anything to do with the query?) and whether the generated output is an accurate summary based on the selected search result. These criteria were ranked from a scale of 0.0 to 1.0, where 0.0 represents “invalid” and 1.0 represents “valid”. Since the number of supported players and the amount of playtime are fields that are typically most important when choosing a board game, the LLM was prompted to ensure that

these values were included in the generated output unless their values were unknown; thus, human evaluation was also used to determine whether the generated output contained this important information. These criteria were ranked on a three-point scale with values 0.0, 0.5, and 1.0, where 0.0 represents “not mentioned”, 0.5 represents “mentioned but is incorrect”, and 1.0 represents “mentioned and is correct”. ROUGE was also used to evaluate the generated output as a summary for the selected search result. It was important to evaluate the summarization using both ROUGE and human evaluation to get a better understanding of the validity of the output.

5 Results

5.1 Human Evaluation

There were four aspects of interest for human evaluation: whether the selected search result is a valid choice for the given query, whether the generated output mentions the number of players that the game supports, whether the generated output mentions the amount of playtime for the game, and whether the output is an accurate summary of the selected search result.

	Valid result for query	**LLM output has # players**	**LLM output has playtime**	**LLM output is accurate**
Avg score	0.86	0.58	0.41	0.58
# samples	20	20	16	20

Figure 1. Human evaluation[2]

As seen in Figure 1, it seems that the model performs decently but with much room for error. It is not surprising that the three criteria that have to do with the LLM output are much lower than the one that has to do with FAISS, since vector search is less of a black box than an LLM is; however, that does not mean that FAISS is perfect either. In fact, 10% of the selected search results were evaluated as not being relevant to the query.

Now moving on to the LLM output judgments, it is notable that one criteria does not have a score for all 20 samples. As mentioned in Section 2, since the number of supported players and the amount of playtime may be unknown for some games, some samples may not have the information in the LLM output nor the search result from which the LLM is referencing. In that case, the criteria is not relevant for evaluation, thereby excluding four

[2] The full annotated chart can be found here.

samples from evaluating whether the LLM output mentions amount of playtime. Regardless, these two criteria have average scores that are quite low. For the criterion about the number of supported players, 11 samples were annotated as mentioning the correct value, 1 sample was annotated as mentioning an incorrect value, and 8 samples were annotated as not mentioning the value at all. For the criterion about the amount of playtime, 6 samples were annotated as mentioning the correct value, 1 sample was annotated as mentioning an incorrect value, and 9 samples were annotated as not mentioning the value at all. For the criterion determining whether the LLM output is an accurate summary of the selected search result, only 2 samples were annotated with a perfect score of 1.0 and 7 samples were annotated with a score less than 0.5.

5.2 ROUGE Scores

ROUGE was used to evaluate the generated output as a summary for the selected search result.

	Avg Precision	**Avg Recall**	**Avg F-Score**
Rouge 1	0.58	0.55	0.52
Rouge 2	0.37	0.37	0.35
Rouge L	0.40	0.39	0.37
# samples	20	20	20

Figure 2. ROUGE[3]

As seen in Figure 2, it is clear that the summarization efforts from the LLM overall are not stellar. Diving deeper into the individual scores, 3 samples had perfect 1.0 score either just for ROUGE-1 precision, for precision across all ROUGE scores, or even all three metrics across all ROUGE scores. This is not a great sign as it indicates that the model is regurgitating some or all of these respective selected search results with no changes. On the other hand, 1 sample had very high scores, with ROUGE-1 precision at 0.97. Upon close inspection, the model summarized the selected search result without copying the exact wording while maintaining key words. Meanwhile, many other samples have extremely low ROUGE scores, with 0.26 as the lowest ROUGE-1 precision score. This suggests that the LLM often hallucinates and pulls information from elsewhere besides the selected search result. In fact, during the earlier testing

[3] Individual ROUGE scores per sample can be found here.

phases, the model often retrieved information from other search results aside from the selected search result. This is likely the cause of these current low scores.

6 Discussion

From observing the results, it is evident that the model is not perfect and has a lot of room for improvement. As mentioned previously in Section 5, some issues with the model stem from poor prompt engineering and hallucinations, seeing as the model often ignores certain instructions and pulls information from outside of the immediate search result or potentially from an external source entirely. When the temperature for the LLM was set to 0.5 or 0.6, interestingly, the results tended to look fairly great for the first few queries, but it would decrease in quality over more queries. When the temperature was set to 0.1 or 0.2, the model often would output only the name of the recommended game and no other information or the output would contain repeated segments. This is an interesting element of LLMs that would be worth exploring further.

For future work, it would be ideal to use a larger model to see if performance improves and to see if other models can perform better than Llama in terms of the issues mentioned above. Since chunking is not used in this project, that may also be useful to tackle the longer game descriptions, which may be a factor for the hallucinations. On the user experience side, originally this project was supposed to be both a recommendation system and a Q&A system that can allow users to ask questions about games, but due to time constraints, the Q&A system idea was not fully fleshed out and thus had to be scrapped.

To answer the question proposed at the end of Section 1, this RAG system is able to provide many good board game recommendations, but there is a long way to go before it can be as effective as a human with domain-specific knowledge and a passion for board games.