Predicting Magical Item Effectiveness in Dungeons and Dragons 5th Edition

A Hunter College CSCI-795 Project Proposal

https://github.com/recursion-ninja/CSCI-795-ML

December 21, 2021

Team:

John Lee
Alex Washburn

Abstract

For our project, we propose classifying Dungeons and Dragons 5th Edition magic items into "power tiers." The purpose of this project is to quantify the efficiency of the various pre-defined magical items as well as predict the effectiveness of novel magic items. Out project will consist of two phases. In the first, we will perform efficacy simulations to generate the observation values for our machine learning models. The second phase requires comparing the application of several machine learning models to best classify the magic items into "power tiers." The intended output of our classifier is to return an integral value $tier \in [0, 20] \subset \mathbb{N}$.

Team Members' Roles

John Lee

- Integrate magic item data into combat simulator.
- Generate observation values from combat simulation.
- Perform EDA to obtain insights of the dataset.
- Build, tune, and evaluate the following models:
  - KNN-Classifier
  - Naive Bayes Classifier
  - Support Vector Machine models.

Alex Washburn

- Collect and collate exhaustive magic item data.
- Clean and normalize the dataset using gameplay keywords.
- Perform EDA to obtain insights of the dataset.
- Build, tune, and evaluate the following models:
  - KNN-Classifier
  - Naive Bayes Classifier
  - Support Vector Machine models.

Motivation

Some work has been done on applying ML to table top role playing games (TTRPGs) in the past [10, 6, 2, 3, 11]. Much of the work revolves around the more tractable problem of selecting appropriate ambient music choices for players to experience based on the current emotional tone of the game [5, 12, 9, 4]. However, the most popular TTRPG, Dungeons and Dragons (D&D) has been used as a difficult test-bed for ML experimentation [7]. This particular previous work, while quite notable, focused entirely on non-combat aspects of D&D, eschewing a core component and past time of D&D; resolving conflict via numerical simulation. In our project we will grapple with this numeric aspect of D&D, focusing on a small subset of the D&D combat system; possession and usage of magical items. To the best of the author's knowledge, the proposed project will be the first serious attempt to apply machine learning to a numerical aspect of D&D. Consequently, there is no *known* previous on which to draw a comparison.

Methodology

We propose producing an effective classifier for D&D magic items by trying KNN-Classifier, Naive Bayes Classifier, and Support Vector Machine techniques. In D&D, the power level of a character or party of characters is often stratified into numerical tiers ranging from 1 to 20. We intend to correspond a magic item's power level to be the maximum D&D party level for which possessing the magic item notably improves suitability when compared to the same scenario without access to the magic item. To do this we propose using a database of existing D&D magic items with their already defined features. Then we will attach 20 observations to each magic item based on the results of combat simulation.

The principle requirement of the AI is to have data to feed it. Unfortunately, the D&D 5e system has many arbitrary elements, and is difficult to quantify. To this end, a combat simulator will be used to generate information for training. The simulation results will be used to generate quantified observation as to the effect the possession of a magic item in a party of character has on their chance of surviving an encounter. These will be represented as one of D&D's primary measurements of power, level difference.

We plan to compile 20 pairs of one character party at the corresponding level ($[1, 20]$) and a combat encounter for the party which ought to be "deadly" on average. For each pair, we will simulate combat 100 times both with access to the magic item and without access. The ratio of the mean survivability with the magic item to the mean over the mean survivability without the magic item will be our observed value. A positive ratio greater than 1.1 will be considered "improved suvivability" and re-encoded to a binary observations.

After this data is generated, we will experiment with different classifier techniques. For the KNN-Classifier and the SVM classifier, we will create 20 binary classifiers of the form "Aids at Level n" where $n \in [1, 20]$. These classifier will be run in parallel and the maximum level classifier which returns a positive result will be treated as the aggregate classifier's resulting "power level" classification. The "power level" classification will be a value $tier \in [0, 20] \subset \mathbb{N}$, where a result of 0 indicates that possession of the magical item does not provide an advantage at any level. When constructing the Naive Bayes Classifier we will tune the model to output the magic item's "power level" directly. We anticipate, the Naive Bayes Classifier will sufficiently model the magic item power level space and can be used to generate more data in the form of theoretical items. These may then be run through combat simulations and fed back in as additional data or used for testing.

Dataset Description

The existing magic item dataset [1] has been pre-complied by members of the 5e Tools open source community and stored in JSON format for public consumption. The dataset we choose is up-to-date as of August 2021. The dataset contains a nested structure which we will parse and flatten to a tabular CSV form. With in the data set are mechanical features such as `"bonusWeapon":"+3"` and `"bonusAc":"+2"`. These features will be incorporated into the first phase of our project to simulate combat with and without access to the magic item's features. Subsequently, these features will be used by the ML techniques above to train classifiers. In total the dataset consists of 1384 samples and 28 features. We will attach 20 additional observation values based on our simulation results.

Timeline

Week	Task(s)
Nov 1	Parse and flatten JSON database
Nov 8	Generate simulated observations
Nov 15	Finalize data set for ML via EDA
Nov 22	Build KNN & SVM classifiers
Nov 29	Build Naive Bayes Classifier
Dec 6	Finalize project video presentation
Dec 13	Final Presentation

Demonstration

The final ML classifier should be capable of taking an input in the form of a magic item, and determining its power level. Additionally, using the Naive Bayes Classifier, it should be capable of randomly generating items that have a specified power level. The type of item should cover any variety of classes, so for class-specific items it would have to be accurate to similar items of that specified power. Generation of items would likely be completely random, as the difficulty of seeking specific bounded results would complicate matters. This could be a stretch goal should the base implementation prove successful. We plan to present our results in the form of a 2 minute video presentation.

Evaluation

For the classification performance, we will be using the standard metrics presented in class, notably precision, recall/sensitivity, and AUC for all three ML techniques described above. Additionally, we will consider the execution time of the classifiers require to output a result, as we expect that the KNN classifier will deficient in this aspect. We will also inspect the confusion matrix manually as it may reveal troublesome areas that a given classifier struggles with and other areas where it excels. It may be possible to create a hybrid classifier combining different techniques and applying the results to their strength and using other techniques to predict in their weaker areas. Due to the lack of *known* previous work, there is no baseline research to compare the project results to. The authors how to collaborate with Professor Raja to determine an *analogous* work on which comparisons can be made. We are considering the recent work on classification of student performance [8] as a potential analogous work.

References

1. 5e.tools. https://github.com/5etools-mirror-1/5etools-mirror-1.github.io/blob/master/data/items.json, 10 2021. GitHub.

2. CAVANAUGH, P. *Machine Learning for Game Master Recommender*. PhD thesis, University of Nebraska at Omaha, 2016.

3. FARIA, F. G., PEREIRA, L. T., AND TOLEDO, C. F. M. Adaptive generation of characters for tabletop role playing games. In *2019 18th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)* (2019), IEEE, pp. 39–46.

4. FERREIRA, L., LELIS, L., AND WHITEHEAD, J. Computer-generated music for tabletop role-playing games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (2020), vol. 16, pp. 59–65.

5. FERREIRA, L., AND WHITEHEAD, J. Mtg: Context-based music composition for tabletop role-playing games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (2017), vol. 13.

6. MACINNES, J. C. The d&d sorting hat: Predicting dungeons and dragons characters from textual backstories.

7. MARTIN, L. J., SOOD, S., AND RIEDL, M. Dungeons and dqns: Toward reinforcement learning agents that play tabletop roleplaying games. In *INT/WICED@ AIIDE* (2018).

8. MUCHUCHUTI, S., NARASIMHAN, L., AND SIDUME, F. Classification model for student performance amelioration. In *Future of Information and Communication Conference* (2019), Springer, pp. 742–755.

9. PADOVANI, R. R., FERREIRA, L. N., AND LELIS, L. H. Bardo: Emotion-based music recommendation for tabletop role-playing games. In *Thirteenth Artificial Intelligence and Interactive Digital Entertainment Conference* (2017).

10. RAMESHKUMAR, R., AND BAILEY, P. Storytelling with dialogue: A Critical Role Dungeons and Dragons Dataset. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (Online, July 2020), Association for Computational Linguistics, pp. 5121–5134.

11. RIEDL, M. O., AND BULITKO, V. Interactive narrative: An intelligent systems approach. *Ai Magazine 34*, 1 (2013), 67–67.

12. RISI, S., AND TOGELIUS, J. Increasing generality in machine learning through procedural content generation. *Nature Machine Intelligence 2*, 8 (2020), 428–436.