# Magic: The Gathering Price Predictor

Initial Report
November 6th, 2019

Michael Ward
Missouri University of Science and Technology
Course: CMP SCI 4300 - Artificial Intelligence
Taken at the University of Missouri - St. Louis

***Abstract*—Wizard's of the Coast's (hereafter referred to as WOTC) "Magic: The Gathering" is the largest Trading Card Game around today. With over 20 billion cards printed from 2008-2016, ten thousand unique cards designed, and with TCGPlayer, the main secondary US market for Magic: The Gathering cards having an es timated annual revenue of 60 million, it's safe to assume there would be some use for a price prediction model [1], [2], [3]. I have attempted to create such a model that uses information about existing cards predicts the price of new cards.**

## I. DATASET

### A. Overview

This project has taken data from `api.scryfall.com` to gather and create the needed dataset that will have the needed inputs and outputs, as discussed a later section. From this source, there are 260,543 individual cards possible, given that the API has each card printing stored as a seperate object.

### B. Creating the Dataset

The first hurdle in creating my dataset was actually downloading the data from Scryfall. This was a major item in itself, as Scryfall separates the API endpoint where data is stored into 1494 pages, and consists of more than 400 MB of JSON data. There were a number of issues getting all of the information - server timeouts, pages not being returned, keys being missing, and the endpoint just straight not working. However, through trial and error and leaving my Python data gatherer running for long periods of time, I eventually was able to create a JSON file with the inputs I thought I would need.

Listing 1. Sample of JSON Data

```
{
 "cards": [
    {
      "name": "Earthshaker Giant",
      "mana_cost": "{4}{G}{G}",
      "cmc": 6.0,
      "colors": [
        "G"
      ],
      "color_identity": [
        "G"
      ],
      "legalities": {
        "standard": "not_legal", ...
      },
      "type_line": "Creature - Giant Druid",
      "reserved": false,
      "set_prices": {
        "Game Night 2019": null
      }
    }, ...
  ]
}
```

| CMC | CI | Formats | Set Date | Type | Price |
|---|---|---|---|---|---|
| 2.0 | 1 | 10 | 0 | 4 | 5.09 |

TABLE I

Explanations of CSV Data and Model Inputs and Outputs

- **CMC.** Converted Mana Cost.
- **CI.** Color Identity. Each color adds 1 to this number, resulting in a range of [0-5].
- **Formats.** Number of formats the card is legal in.
- **Set Date.** How many years ago the cheapest printing of the card was printed. For example: if a card's cheapest reprint was in 2017, this value will be 2.
- **Type.** This is a value that represents the major typelines on the card. I developed a system where each type is represented by a different number.
- **Price.** Price of the card's cheapest printing. (The output of the Model.)

### C. Working with CSV

After I had my the JSON data compiled, I created another Python program to turn the relevant information into a CSV file. After creating the CSV file, I uploaded it to my personal github account so my google collab document could readily access it.

### D. Honing the dataset

Firstly, as the data was sorted by most recent print, I randomized the order of my dataset before running it through any models. Then, I normalized each input value to have an average of 0 and a standard deviation of 1 so later, weights could be applied to properly determine the impact each input has on the final result.

Then, as I began working with my dataset and running it through basic models, I made a determination that any cards on the WOTC Official Reserved List - a group of cards WOTC has declared will never be reprinted - drove the price far too high. Cards ranging in the hundreds or sometimes thousands of dollars completely wrecked the model's ability to predict prices accurately. So instead of factoring this outlier in, I decided to remove them from my dataset completely, as the purpose of this model is to predict the value of new cards, and WOTC has stated they will not add any new cards to the Reserved List [5].

Another factor I discovered as I was honing my dataset was that another input I thought may have had an impact on price - whether or not it was a able to be played as a Commander in the Commander format - seemed to not have an impact on on the price at all, as there seemed to be little to no correlation between its eligibility to be a Commander and its price. Therefore, I removed this from my CSV file and from my input set.

## II. Model Implementation and Testing

### A. Data Splitting

Before implementing the model, I split the dataset will be split into 3 sections:

- 70 percent of the data for training
- 15 percent of the data for testing
- 15 percent of the data for validation

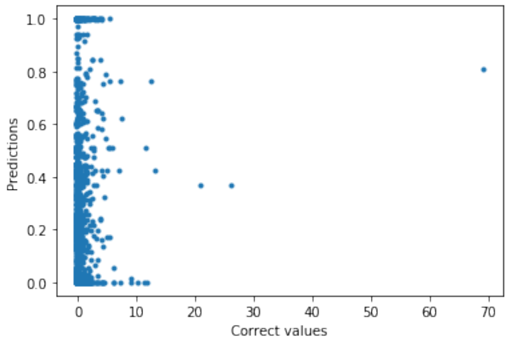
### B. Implementation and Testing

I created my model in Google Colab. Starting with a basic model with a few layers, 50 epochs, and a linear activation with the final layer of neurons. I fiddled with the model a lot, going up to 150 epochs, adding more layers, trying to determine a good model. Finally I normalized my output and used a sigmoid activation on the final layer, and that seemed to provide a good result as it dropped MAE to .28, which is the lowest I had seen up to that point.

## III. Tests and Evaluation

Even though I reached a low MAE, the amount of fiddling I did still did not seem to be enough, as shown in Figure 1. Unless I determine some way before the final report to improve the model that I have not yet determined, it must be concluded that the inputs are not valid to accurately predict card prices. I will be doing more research and testing between now and December as I believe this is not the case.

Fig. 1. Test Results




## IV. Wrapping Up

Over the course of this project, I faced quite a few challenges, some documented above. Other challenges I faced were less technical in nature, such as locking myself out of my GitHub account, not being able to install the environment properly, or numpy adding columns to my dataset for no apparent reason. Those, at least, have all been overcome.

I'm disappointed that I was unable to reach a satisfactory model for price prediction, as this is something I was hoping I would be able to use myself. Which is one reason why before the final report is due, I will be doing my best in researching and reaching out to those more experienced to try and create a more satisfactory model.

## References

[1] https://magic.wizards.com/en/content/magic-25th-anniversary-page-facts-and-figures
[2] https://magic.wizards.com/en/articles/archive/25-random-things-about-magic-2009-02-16
[3] https://www.owler.com/company/tcgplayer
[4] https://scryfall.com/docs/api/cards/all
[5] https://mtg.gamepedia.com/Reserved_List#2002_revision