

Databases and Advanced Data Techniques Mid-Term coursework

LINK: <https://hub.labs.coursera.org:443/connect/sharedydpztia?forceRefresh=false&path=%2F%3Ffolder%3D%2Fhome%2Fcoder%2Fproject>

Stage 1: Find and critique a dataset

Choosing a source of open data

For my dataset, I will be using some data on Pokemon taken from Bulbapedia. The data is open and to there are no normalised, relational models for my data set.

Assessing the dataset

Quality

The data is reliable, as it is extracted from the game of Pokemon itself, where most of the information is easily available and accessible for users to see.

Detail

The data is detailed, containing most of the information needed to process information.

Documentation

The data is clear in terms of meaning as long as a basic understanding of Pokemon terms is provided. This is easily supplemented by the information available in Bulbapedia, the source of the data.

Interrelation

As the data for Pokemon and their moves are originally separated in Bulbapedia, connecting the data sets is indeed useful for a clearer understanding of Pokemon data. It is also easy to accomplish, since there are keys that can link the two datasets together. This would be further improved by the data about each individual move being connected.

Use

It would be possible to observe patterns in the Pokemon, as well as compare and calculate different sets of numerical data, for example, the damage dealt by the Pokemon. However, it would be difficult to get the dataset to produce numbers on the damage dealt, since it would be necessary to factor in additional datasets for types, exceptions, and all other factors that affect damage calculations. This would be difficult to represent and store in the currently available

Discoverability

It was relatively easy to find open data in my chosen domain of Pokemon. I went to Bulbapedia to find the data, more specifically from their content covering:

“List of Pokémon by National Pokédex number”

https://bulbapedia.bulbagarden.net/wiki/List_of_Pok%C3%A9mon_by_National_Pok%C3%A9dex_number

As well as all the individual Pokemon's Bulbapedia pages that covers their stats, the moves they can use as of Generation 7, and the stats of those moves.

There are not a lot of other alternatives that are clearly open source and carry all the data needed for a holistic analysis.

Licensing

This dataset was found on Bulbapedia, the community-driven Pokémon encyclopedia. It is licensed with the *Creative Commons Attribution-NonCommercial-ShareAlike 2.5 Generic (CC BY-NC-SA 2.5)*, which means that I am free to share and adapt the materials as long as I give proper credit to Bulbapedia, I do not use it for commercial purposes, and I also license the resulting work under the same copyright.

Explaining my interest in the dataset

What is a Pokemon?

Pokemon are digital creatures that exist in the game Pokemon. They possess what players call stats(particular numerical values related to each Pokemon), which define different properties such as hp(health points), attack, and speed. These stats affect how each Pokemon is played, in combination with some other mechanics that exist in the game, like abilities, moves and types.

Why is this an interesting dataset?

This is an interesting dataset to me, as I am a fan of Pokemon, but I never really took the time to analyse the deeper mechanics behind the game itself. One such mechanic is called Pokemon battling where a team of Pokemon face off against another's team(which may be AI controlled or another actual player). Being a rather casual fan, I often find it difficult to completely comprehend how a Pokemon battle works, even though I frequently participate in games on Pokemon Showdown(A Pokemon battle simulator). My research into the subject matter showed that a lot of data in the game can be analysed to answer queries on Pokemon battling.

Why did I choose this dataset?

I chose the dataset containing information mainly concerning Pokemon battles, which would be information on Pokemon stats, their types, abilities and their moves. However, due to abilities being something that is difficult to process using a database and a simple web application, I chose to exclude it from the data. This leaves me with largely pure numerical values to work with.

Future of this dataset and web app

If further worked upon and with other additional resources, I think it is possible to come up with a program that is able to answer all the different questions pertaining to Pokemon battling, such as the best possible combination of moves for a Pokemon.

Some questions that can be answered with this dataset

- 1) What Pokemon has the most moves available?
- 2) Which Pokemon has the highest speed value?
- 3) What is the potential strongest moves for a Pokemon like a Lapras?

Stage 2: Model your data

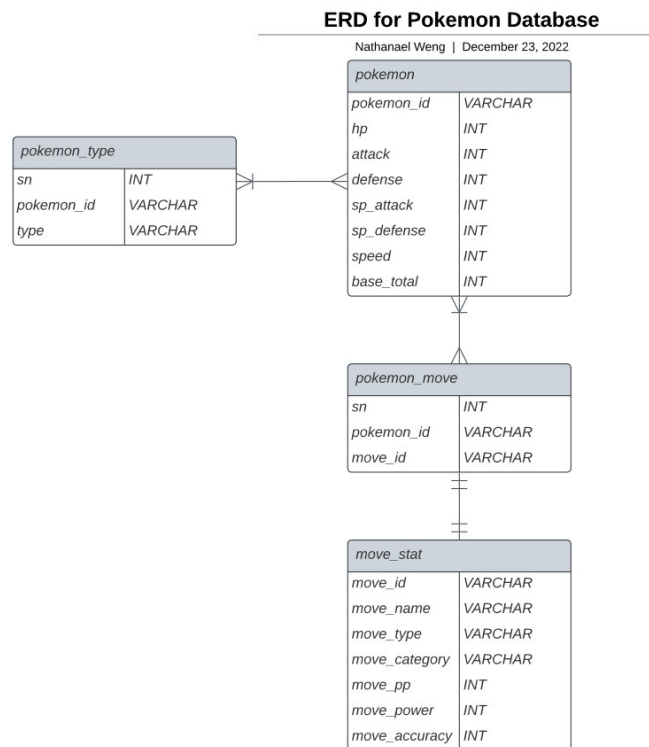


Fig 1.0 E/R Diagram

Explaining table names and their contents

Entity Names	Description
pokemon	Table containing name(pokemon_id) and stats(hp, attack, defense, sp_attack, sp_defense, speed, base_total) of the Pokemon
pokemon_move	Contains a pokemon name that links to each move a pokemon can make, using a numeric key to uniquely identify each row
move_stat	Name of the move, move's type, move's category, the pp of the move, the power of the move, and the move's accuracy.
pokemon_type	Name of Pokemon and its type(s)

Conversion to normal form

First Normal Form(1NF)

The definition the 1NF is that a relation must have a single valued attribute(no composite values like a list of items e.g. [1,2,3]). Since a Pokemon can have multiple types and multiple moves, I had to separate the types and the moves from the other Pokemon attributes, to reduce redundancy. I gave them a table with an alternative primary key to help link them back to the Pokemon data. For types, I could have presented the two types in Pokemon as two separate columns, however, I opted to present it as a single column of types, in order to better enable future scalability. This opens up the possibility of adding a third type for each individual Pokemon.

Second Normal Form(2NF)

The definition of 2NF is that it must first be in 1NF and every non-prime attribute of the table is fundamentally dependent on the primary key. This means that every other key has to be directly derived from the primary key and not something that is directly derived from other attributes. In order to achieve this, all data was split up into their functional dependencies. The pokemon table only contains the name and the stats, and the moves table only contains the name and attributes that depend on the move name.

Third Normal Form (3NF)

The definition of 3NF is that it must be in 2NF and every non-prime attribute is not transitively dependent on every key. This means that every other key has to be directly derived from the primary key and not something that is indirectly derived from the primary key. Due to the nature of each of the various entities in the Pokemon dataset, by splitting into 2NF, 3NF has already been achieved due to the way attributes are grouped.

3NF was sufficient for my dataset as it is impractical to further convert the data set into other normal forms, as it would only make it difficult to comprehend the information from the database without processing it first.

Stage 3: Create the database

Make a database called poke_info

```
CREATE DATABASE poke_info;
```

```
USE poke_info;
```

pokemon table

```
CREATE TABLE pokemon(  
  pokemon_id varchar(32) NOT NULL,  
  hp int NOT NULL,  
  attack int NOT NULL,  
  defense int NOT NULL,  
  sp_attack int NOT NULL,  
  sp_defense int NOT NULL,  
  speed int NOT NULL,  
  base_total int NOT NULL,  
  PRIMARY KEY (pokemon_id)  
);
```

Loading instance data using individual CSV

```
LOAD DATA INFILE '/home/coder/project/mid-term/poke-info-db/data/pokemon.csv'  
INTO TABLE pokemon  
FIELDS TERMINATED BY ','  
OPTIONALLY ENCLOSED BY '"'  
LINES TERMINATED BY '\n'  
IGNORE 1 LINES;
```

pokemon type table

```
CREATE TABLE pokemon_type(  
  sn int NOT NULL,  
  pokemon_id varchar(32) NOT NULL,  
  type varchar(16) NOT NULL,  
  PRIMARY KEY (sn)  
);
```

Loading instance data using individual CSV

```
LOAD DATA INFILE '/home/coder/project/mid-term/poke-info-db/data/pokemon_type.csv'  
INTO TABLE pokemon_type  
FIELDS TERMINATED BY ','  
OPTIONALLY ENCLOSED BY '"'  
LINES TERMINATED BY '\n'  
IGNORE 1 LINES;
```

all moves in pokemon table

```
CREATE TABLE moves(  
  move_id varchar(64) NOT NULL,  
  move_name varchar(64) NOT NULL,  
  move_type varchar(16) NOT NULL,  
  move_category varchar(16) NOT NULL,  
  move_pp int,  
  move_power int,  
  move_accuracy int,  
  PRIMARY KEY (move_id)  
);
```

Loading instance data using individual CSV

```
LOAD DATA INFILE '/home/coder/project/mid-term/poke-info-db/data/moves.csv'  
INTO TABLE moves  
FIELDS TERMINATED BY ','  
OPTIONALLY ENCLOSED BY '"'  
LINES TERMINATED BY '\n'  
IGNORE 1 LINES;
```

pokemon move set table

```
CREATE TABLE pokemon_move(  
  pokemon_id varchar(32) NOT NULL,  
  move_id varchar(64) NOT NULL  
);
```

Loading instance data using individual CSV

```
LOAD DATA INFILE '/home/coder/project/mid-term/poke-info-db/data/poke_move.csv'  
INTO TABLE pokemon_move  
FIELDS TERMINATED BY ','  
OPTIONALLY ENCLOSED BY '"'  
LINES TERMINATED BY '\n'  
IGNORE 1 LINES;
```


adding foreign key constraints

```
ALTER TABLE pokemon_type  
ADD FOREIGN KEY (pokemon_id) REFERENCES pokemon(pokemon_id);
```

```
ALTER TABLE pokemon_move  
ADD FOREIGN KEY (pokemon_id) REFERENCES pokemon(pokemon_id);
```

```
ALTER TABLE pokemon_move  
ADD FOREIGN KEY (move_id) REFERENCES moves(move_id);
```

```
SELECT pokemon_move.*  
FROM pokemon_move  
LEFT JOIN moves ON moves.move_id = pokemon_move.move_id  
WHERE moves.move_id IS NULL;
```

[localhost:3000/](#)


Data on Pokemon

[Most Moves Available](#)[Top 5 in speed](#)[Lapra's Strongest Move\(s\)](#)

Ten Pokemon names!

- Piloswine
- Metapod
- Graveler
- Diancie
- Torracat
- Bewear
- Charjabug
- Probopass
- Wooper
- Zweilous

Home Page

[localhost:3000/mostmoves](#)

Pokemon with the Most Moves

Pokemon Name:	Number of Moves:
Mew	322

Most Moves Page

Top 5 in speed

Pokemon Names:	Speed:
Deoxys	180
Ninjask	160
Pheromosa	151
Alakazam	150
Aerodactyl	150

Strongest Move(s) for Lapras

Move Name:	Type:	Category:	PP:	Accuracy:	Power:
Giga Impact	Normal	Physical	5	90	150
Hyper Beam	Normal	Special	5	90	150
Skull Bash	Normal	Physical	10	100	130
Double-Edge	Normal	Physical	15	100	120
Future Sight	Psychic	Special	10	0	120