POKÉMON TYPE CLASSIFIER

CSI 4810: PROJECT 2

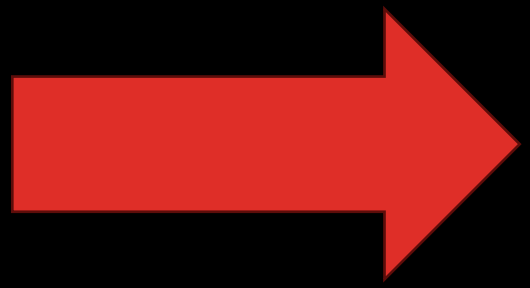
Jason Kauppila

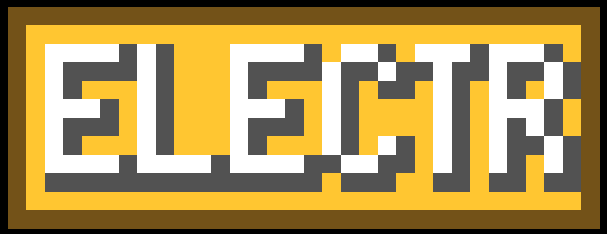
DOMAIN KNOWLEDGE

- Pokémon are fictional creatures from the video games series known by the same name.
- As of today, there are 1025 different species of Pokémon that have been discovered.
- Each Pokémon species has one or two different types.
  - There are 18 possible regular types that a Pokémon species can have:
    - Fire, Water, Grass, Electric, Ice, Fighting, Poison, Ground, Flying, Psychic, Bug, Rock, Ghost, Dark, Dragon, Steel, Fairy, Normal
- Information about each Pokémon species is stored in the Pokédex.
  - Each species has a short textual entry known as “flavor text”.

PROJECT AIM

- Build a classifier model to do the following task:
  - Given a short description of a Pokémon
    - Predict the type of the Pokémon





"When several of these Pokémon gather, their electricity can build and cause lightning storms."

MODEL USEFULNESS

- Usefulness in the fictional realm:
  - Being able to predict the type of Pokémon could be useful for Pokémon Trainers in the world of Pokémon, especially those trying to collect data for the Pokédex.
- Usefulness in the real world:
  - Designers of new Pokémon for the video game series may find the classifier model useful for assigning types to new concepts that they have for Pokémon.
    - If the designer already has an idea in mind for the type of a Pokémon, it would be useful to see if the Pokémon's description is consistent with its typing.
      - Otherwise consumers may find it difficult to identify the type of a Pokémon, potentially impacting the reception of the new design.
  - The concept of elemental beasts may generalize to fictional worlds beyond that of Pokémon.
    - The model may serve as a useful tool for writers whose work contains elemental beasts.

DATA COLLECTION

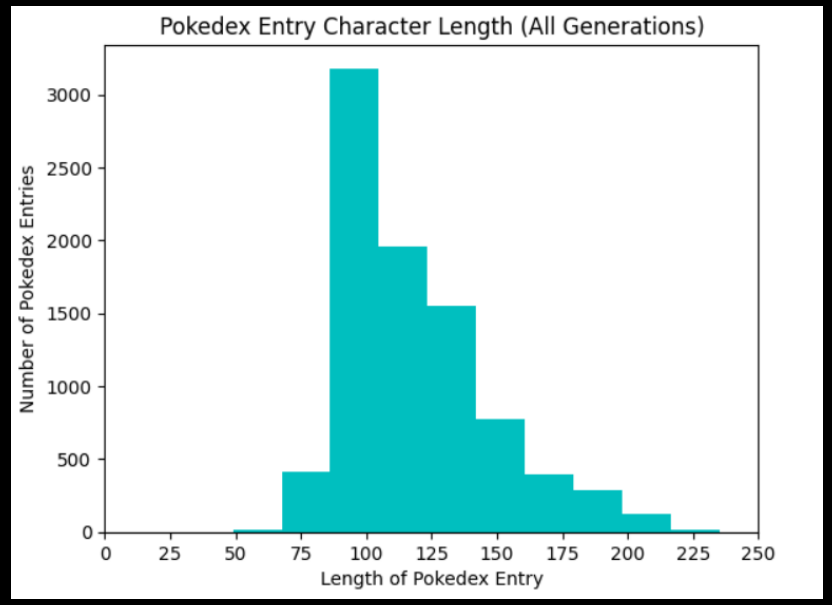
- Information about each Pokémon was scraped from pokemondb.net.
- Web scraping was done using the Requests and Beautiful Soup Python libraries with Jupyter Notebook being used the programming environment.
  - From the webpage listing all the Pokémon species (https://pokemondb.net/pokedex/all), links to the pages for each individual Pokémon were retrieved.
    - The following data was retrieved from the pages for each Pokémon:
      - Pokémon Name
      - National Dex Number
      - Pokémon Type(s)
      - Pokédex Flavor Text
- Prior to converting the data into a Pandas DataFrame in order to save as a CSV file:
  - Duplicate flavor text entries within a given Pokémon species were removed.
  - One hot encoding was applied to the types for each Pokémon.

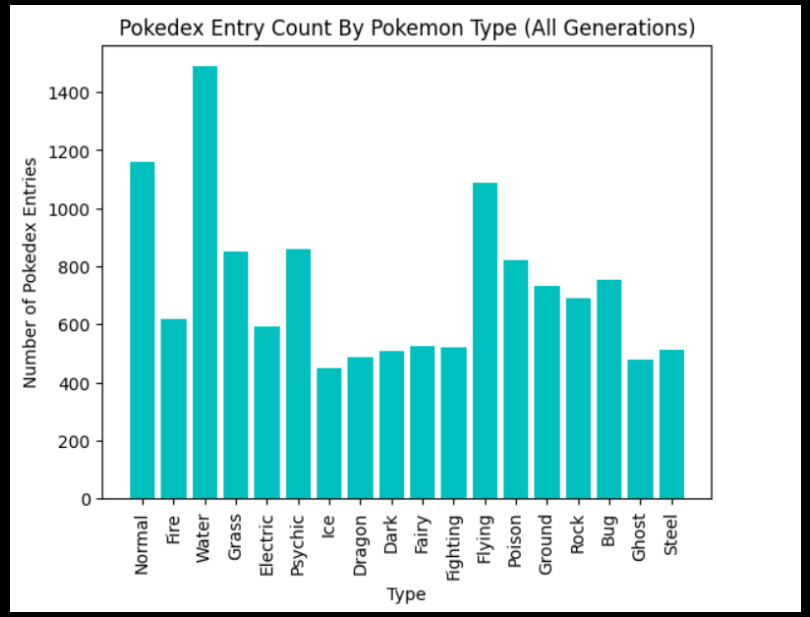
DATA CLEANING

- Upon reviewing the scraped data, it was discovered that some Pokémon had more than two types, which should not be possible.
  - It was discovered that this was caused by a Pokémon species having multiple forms where each form took on a different type.
  - All the Pokémon who have multiple forms were manually reviewed for type changes, and the corresponding rows in the dataset were corrected.
    - The flavor text corresponding to each form was associated with the correct typing.
    - If the flavor text did not specify a specific form of a Pokémon, the default form and its typing were used.
- Some Pokémon of different species had the exact same flavor text despite having different types.
  - Buzzwole, Pheromosa, Xurkitree, Celesteela, Kartana, Guzzlord
  - "Although it's alien to this world and a danger here, it's apparently a common organism in the world where it normally lives."
    - These 6 rows were removed.

CURRENT STATE OF THE DATA

- At this point, there are 8724 flavor text entries associated with Pokémon types.
- Next, simple visualization of the data was done with the additional help of the Matplotlib library.
  - Visualize the distribution of flavor text length.
  - Visualize the frequency of each Pokémon type.



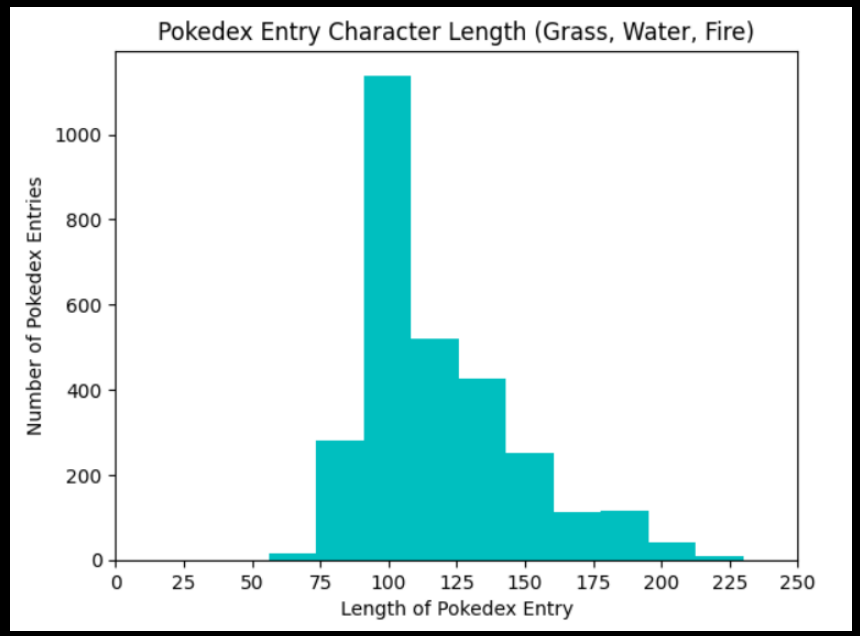


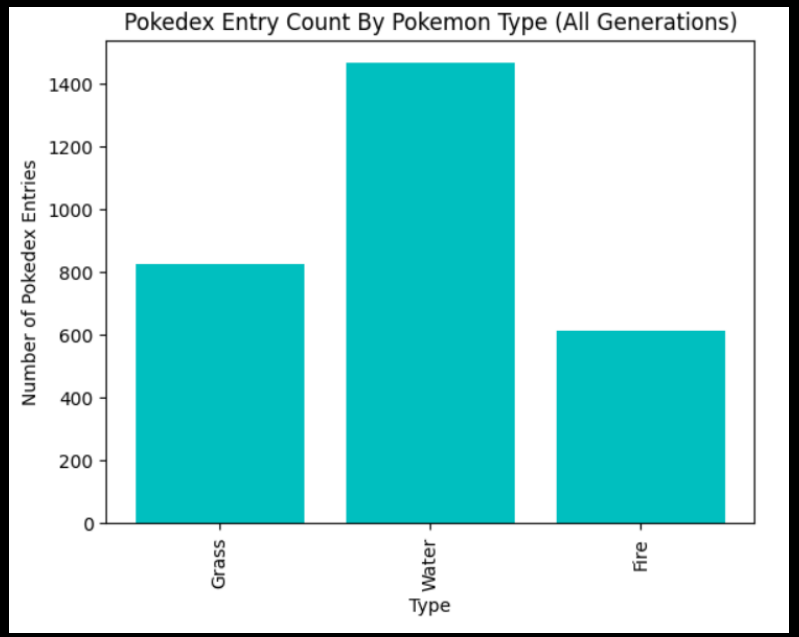
LIMITING THE PROJECT SCOPE

- Due to the large number of different Pokémon types and the fact that Pokémon can have up to two different types, it seemed unlikely that the model would produce good results with the current dataset.
  - Overlapping concepts between Pokémon types.
  - For Pokémon with multiple types, Pokédex entries may reflect one typing more strongly than the other.
  - Pokédex entries may contain information unrelated to typing.
  - Limited number of Pokédex entries considering there are 153 double type possibilities plus 18 single type possibilities.
- New Formulation:
  - Given three Pokémon types, predict which one of them would be assigned to a Pokémon with a given description.
    - For simplicity, we will use three most of the iconic types:
      - Grass
      - Water
      - Fire

CLEANING THE DATA SUBSET

- From the entire dataset, only Pokémon with the type grass, water, or fire were kept.
  - For Pokémon with two types, the second type was discarded.
    - If both of the Pokémon's types were grass, water, or fire, their entries were completely removed.
  - 2908 data entries remained.





MODEL USED

- Flavor text converted to float vectors (384 dimensions) using the text embedding model "all-MiniLM-L6-v2" with the Sentence Transformers library from SBERT.net.
- Use these vectors to build 3 classifier models with the Scikit-Learn library:
  - Naïve Bayes
  - Logistic Regression
  - Neural Network (Multi-Layer Perceptron Classifier)
- Evaluate each classifier model using a stratified 80-20 train-test split.
  - Evaluation metrics:
    - Weighted F1-Score
    - Balanced Accuracy
      - (The average of the recall obtained for each class.)
  - Determine the best model to use for this problem.

EXPECTED RESULTS

- It is expected that the Neural Network model will produce the best results.
  - Naïve Bayes and Logistic Regression are linear classifiers.
- When choosing between only three classes (Pokémon types), good model performance is expected.
  - Estimate for the expected performance is roughly 81% balanced accuracy.

ISSUES FACED

- Complexity of the Problem
  - Initially consulted Scikit-Learn's documentation for information regarding multilabel classification.
  - Ultimately addressed by reducing the project scope to a subset of Pokémon types.
- Data Imbalance
  - Consulted an article from Data Science Horizons regarding working with imbalanced datasets.
    - Tried to use random undersampling and random oversampling with the Imbalance-learn library to address the imbalance.
- Web Scraping Problems
  - Initial project aim was to build a star rating classifier for product reviews.
    - Captchas prevented scraping more than 90 reviews, which was not a large enough dataset.

MODEL RESULTS (NAÏVE BAYES)

- Implemented using Scikit-Learn's GaussianNB with default parameters.
- Using Unbalanced Dataset:
  - Weighted F1-Score:
    - 0.8428
  - Balanced Accuracy:
    - 0.8574
- Using Random Undersampling:
  - Weighted F1-Score:
    - 0.8288
  - Balanced Accuracy:
    - 0.8265
- Using Random Oversampling:
  - Weighted F1-Score:
    - 0.8372
  - Balanced Accuracy:
    - 0.8471

MODEL RESULTS (LOGISTIC REGRESSION)

- Implemented using Scikit-Learn's LogisticRegression with the following parameters:
  - Max Iterations: 700
  - Random State: 13
  - Solver: 'sag'
  - C Value: 8 (determined through experimentation)
  - Other Parameters: Default
- Undersampling was not done due to poor performance.

Unbalanced Dataset		
C Value	Weighted F1 Score	Balanced Accuracy
1	0.8509	0.8284
2	0.8562	0.8351
4	0.8619	0.8465
8	0.8601	0.8542
16	0.8652	0.8518
32	0.8566	0.8428

Oversampling		
C Value	Weighted F1 Score	Balanced Accuracy
1	0.8457	0.8401
2	0.8523	0.8476
4	0.8471	0.8424
8	0.8573	0.8533
16	0.8503	0.8454
32	0.8463	0.8372

MODEL RESULTS (NEURAL NETWORK)

- Implemented using Scikit-Learn's MLPClassifiers with the following parameters:
  - Max Iterations: 700
  - Random State: 13
  - Hidden Layer Size: [600] (determined through experimentation)
  - Other Parameters: Default (default activation function is 'relu' and default solver is 'adam')
- Undersampling was not done due to poor performance.

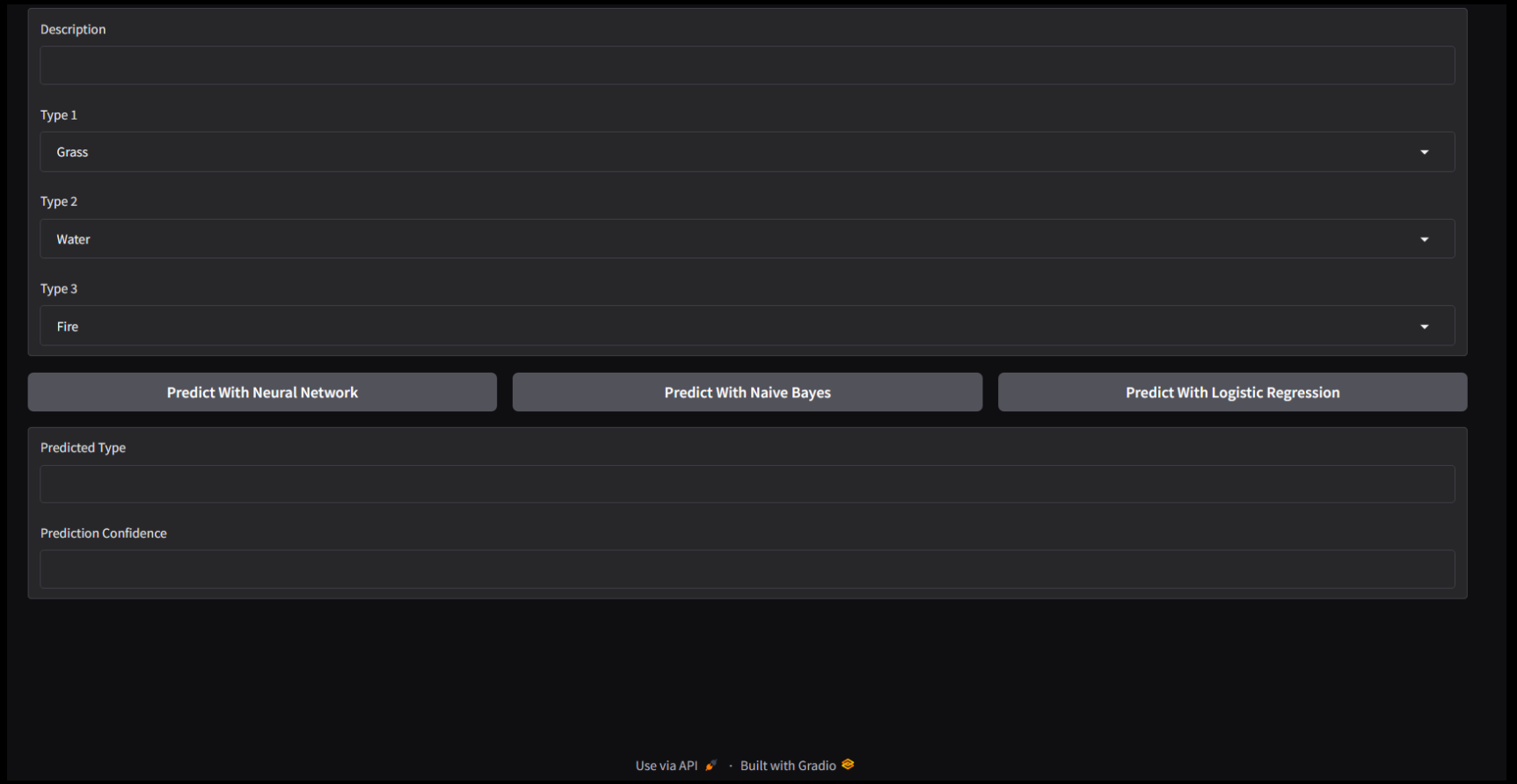
Unbalanced Dataset		
Hidden Layer Size	Weighted F1 Score	Balanced Accuracy
[300]	0.8845	0.8758
[400]	0.8847	0.8791
[500]	0.8881	0.8798
[600]	0.8931	0.8818
[700]	0.8861	0.8754

Oversampling		
Hidden Layer Size	Weighted F1 Score	Balanced Accuracy
[300]	0.8898	0.8825
[400]	0.8863	0.8771
[500]	0.8881	0.8807
[600]	0.8932	0.8834
[700]	0.8846	0.8753

MODEL RESULTS (CONCLUSIONS)

- Random oversampling did not significantly impact model performance.
  - Comparable results are obtained using the base unbalanced dataset.
  - For this reason, random oversampling is not done for the models in the Gradio demo application.
- Model Parameters:
  - For the Logistic Regression Classifier, the best C value was 8.
  - For the Neural Network Classifier, the best hidden layer size was [600].
- Performance:
  - The Neural Network Classifier performed the best.
  - The Naïve Bayes Classifier performed comparably to the Logistic Regression Classifier.

GRADIO DEMO APPLICATION



SOURCES (INFORMATION AND TOOLS)

- Slide 2:
  - https://bulbapedia.bulbagarden.net/wiki/Type
  - https://pokemondb.net/pokedex/all
- Slide 3:
  - https://pokemondb.net/pokedex/pikachu
- Slide 4:
  - https://pokemondb.net/
  - https://pypi.org/project/requests/
  - https://beautiful-soup-4.readthedocs.io/en/latest/
  - https://www.python.org/
  - https://jupyter.org/
  - https://pandas.pydata.org/docs/
- Slide 5:
  - https://pokemondb.net/
- Slide 6:
  - https://pokemondb.net/pokedex/buzzwole
- Slide 7:
  - https://matplotlib.org/
- Slide 10:
  - https://www.sbert.net/docs/sentence_transformer/pretrained_models.html
  - https://scikit-learn.org/stable/index.html
- Slide 12:
  - https://scikit-learn.org/stable/modules/multiclass.html
  - https://datasciencehorizons.com/handling-imbalanced-datasets-in-scikit-learn-techniques-and-best-practices/
  - https://imbalanced-learn.org/stable/index.html
- Slide 13:
  - https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html
- Slide 17:
  - https://www.gradio.app/

SOURCES (IMAGES)

- Slide 2:
  - https://archives.bulbagarden.net/media/upload/c/cb/SV_type_mural.png
- Slide 3:
  - https://www.serebii.net/swordshield/pokemon/025.png
  - https://www.serebii.net/pokedex-bw/type/electric.gif

THANK YOU

Feel free to leave feedback or comments!