# Sea Turtle Conservation Project

Devin DuBois

University of Florida

EEL3872 - AI Fundamentals

Prof. Diego Alvarado

March 22, 2025

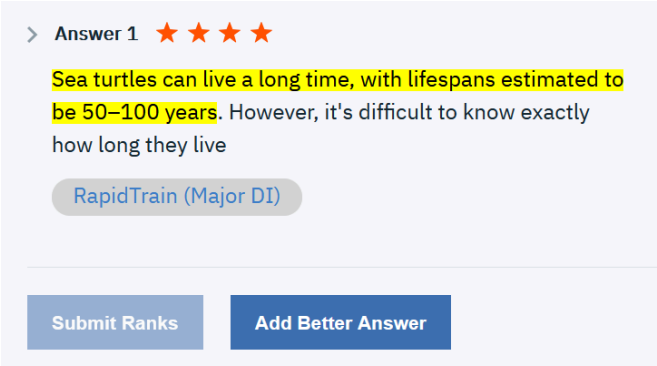
# Abstract

This report outlines the training process and key findings for CogUniversity's Cammy ChatBot. The training was divided into three parts: Sea Turtle Basics, Sea Turtle Weight Prediction, and Sea Turtle Species Identification. This describes how data was allocated across these sections and examines its impact on Cammy's performance. Although the bot has proven effective as an educational resource, additional training is needed to achieve its original mission of "saving the turtles."

# Table of Contents

# Part 1 - Sea Turtle Question Training

Training the bot Cammy was an interesting experience.  It was very tedious to go through all 40 questions but in the end, I provided a lot of useful data for the bot to answer different questions about sea turtles.  I started with having to input a lot of new questions, but as time went on I was able to use the bot's responses more and more.




Example question

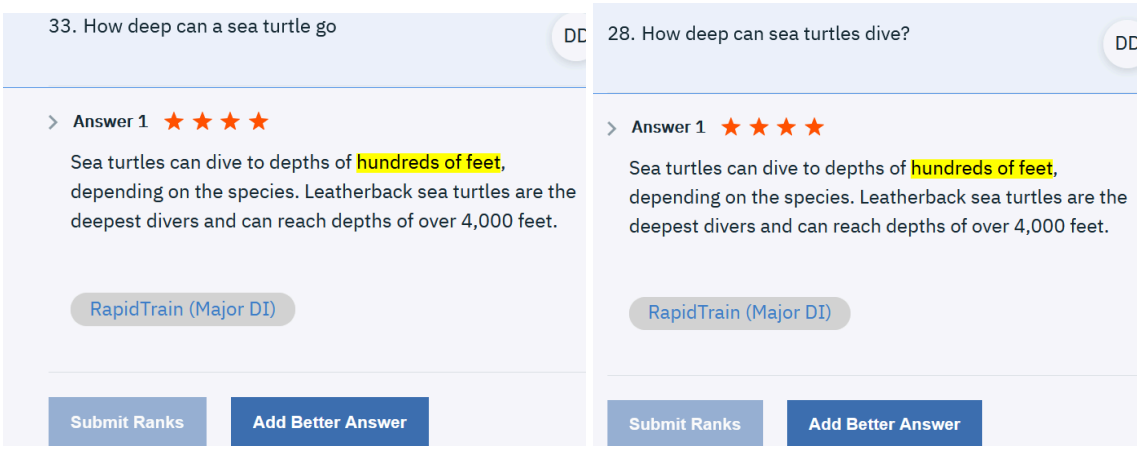

Training data set sample

## Target Audience

I think the best way to train Cammy for a specific target audience is to ask more fun or interesting questions about sea turtles. Some questions I would add to make Cammy more interesting to the target audience are more fun questions, like “What colors can sea turtles Be” or “How many sea turtles are in the World”.

## Training Efficiency

An efficient way to train a CogBot to answer questions is to develop new questions that a new user would think of. There is a wide variety of questions, so polling users for new questions would increase the variety of possible questions and answers and improve performance.




Two different questions with the same answer

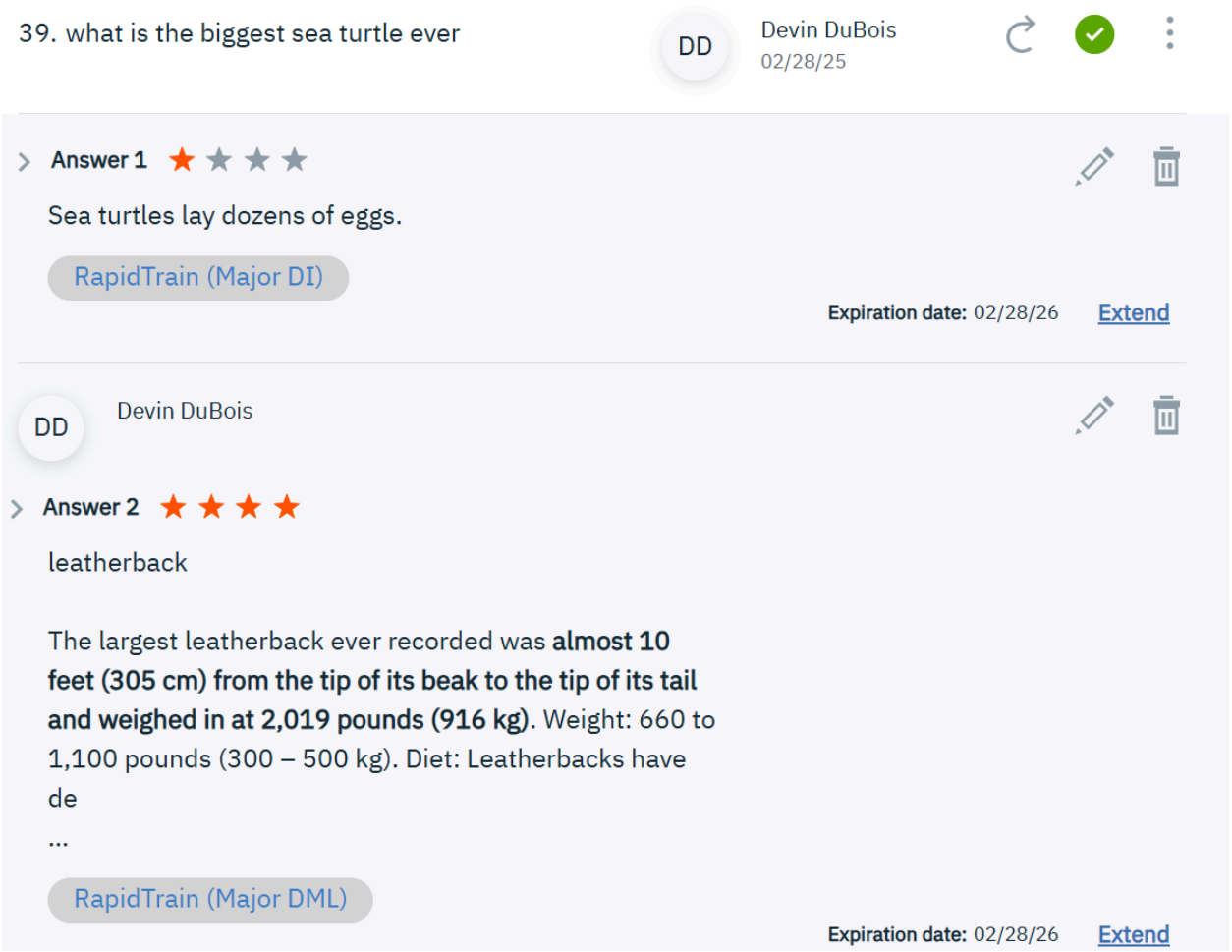
To improve machine learning models to give better predictions, one foolproof method is to provide more data. The more data a machine learning model has to work with, the more fine-tuned the inferences can be about future predictions.

My cogbot learned pretty quickly, as after only 10-15 questions, it provided a sufficient answer a majority of the time. Some of the questions were worded differently but essentially asked the same thing. Some questions had similar words as previous questions but were asking a different question than previous questions, which confused the model.

A cogbot needs data to learn, and cannot learn instantly. The process can be sped up by having multiple people work on it or by the machine providing correct answers. The more data it has, the less often the human needs to add new questions, taking up time. Therefore as the bot gets trained it learns faster.

## Model Confusion

You can confuse Cammy by providing incomplete or incorrect answers and ratings. You can also confuse Cammy by providing a different answer when the answer is already in the corpus, making it think the other answer would be wrong.  To avoid confusing Cammy, it is recommended to re-use questions when possible.




Reused answer for a different question

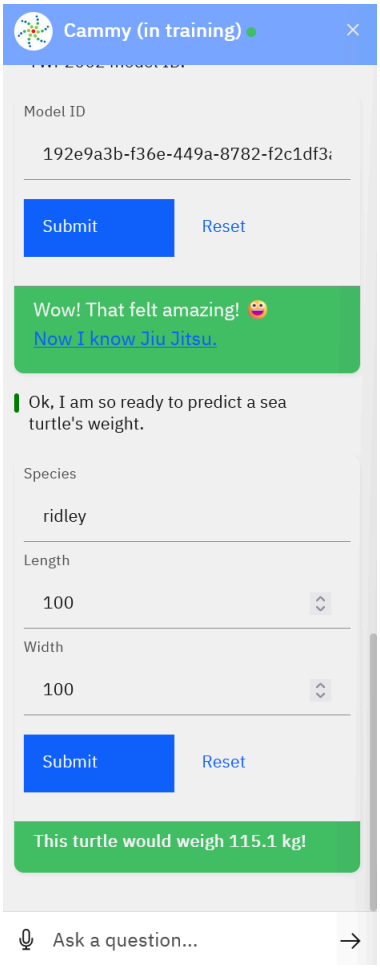
# Part 2 - Regression and Data Visualization

Part 2 showed me data visualization with Google Data Studio, dealing with data in Excel, and training an AI model with lots of data. My takeaway is that lots and lots of data is needed for artificial intelligence. There are a lot of interesting ways to make use of it and compare data against each other to get interesting results.

In this section, regression is used to predict the weight of turtles given it's species, carapace length, and carapace height.

## Auto AI in the industry

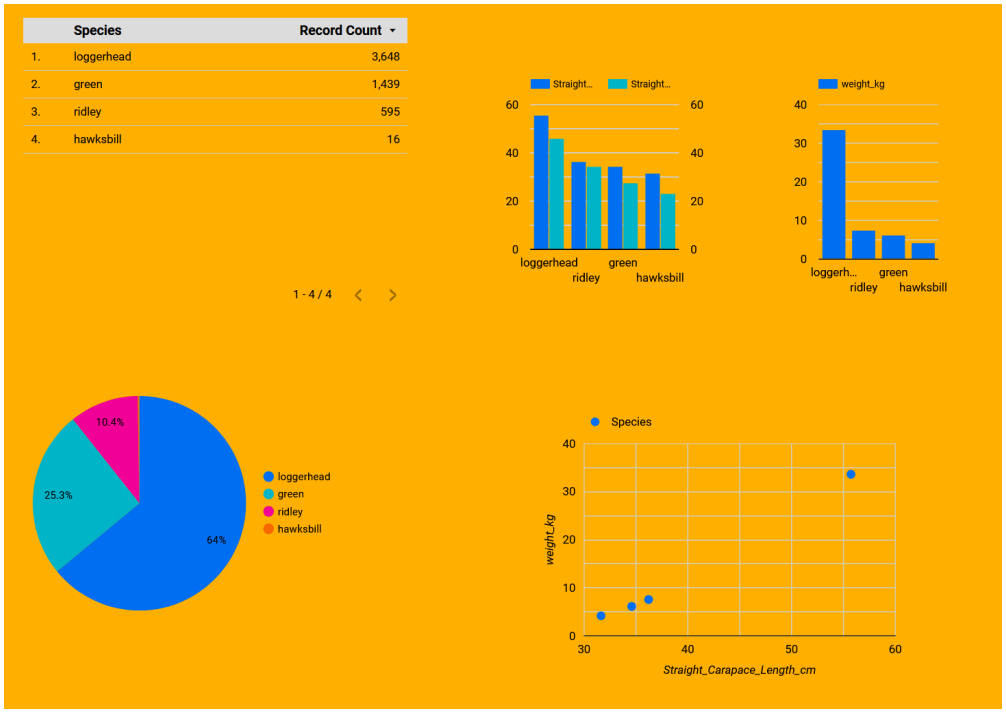
Auto AI is overall good for the industry because it takes away a lot of overhead from manually comparing data and coming to a conclusion. It saves a lot of man-hours manually reviewing data and can show some insights that humans could not come to without it.



Google data studio is a very useful tool that I can use to quickly show plots and graphs without using a programming language like R or Python. If I need to create a report from data I can throw a quick report together pretty quickly. Cammy's auto AI is also useful because it trains models without any direction or instructions just from a dataset.

## Google Data Studio

I explored the data by creating varying visualizations of the data. There are 4 different species with varying amounts of records. The bar charts show the different Carapace width and height averages, as well as the weight in kilograms. The pie chart shows the percent of the dataset each species is. The scatterplot shows the weight in kg compared to the carapace length.




Data visualization

Using the tools learned, I want to add more species and more data to categorize which species it is just based on the weight and carapace dimensions. I think this would be an interesting way to flip how data is used to predict outcomes.

## Improving Regression Models

We can improve regression machine learning models using different techniques such as cross-validation or bagging, and by providing more data. The more data is

provided the more accurate the model will be, especially for underrepresented data such as the hawksbill turtle. Training machine learning models requires a lot of data.

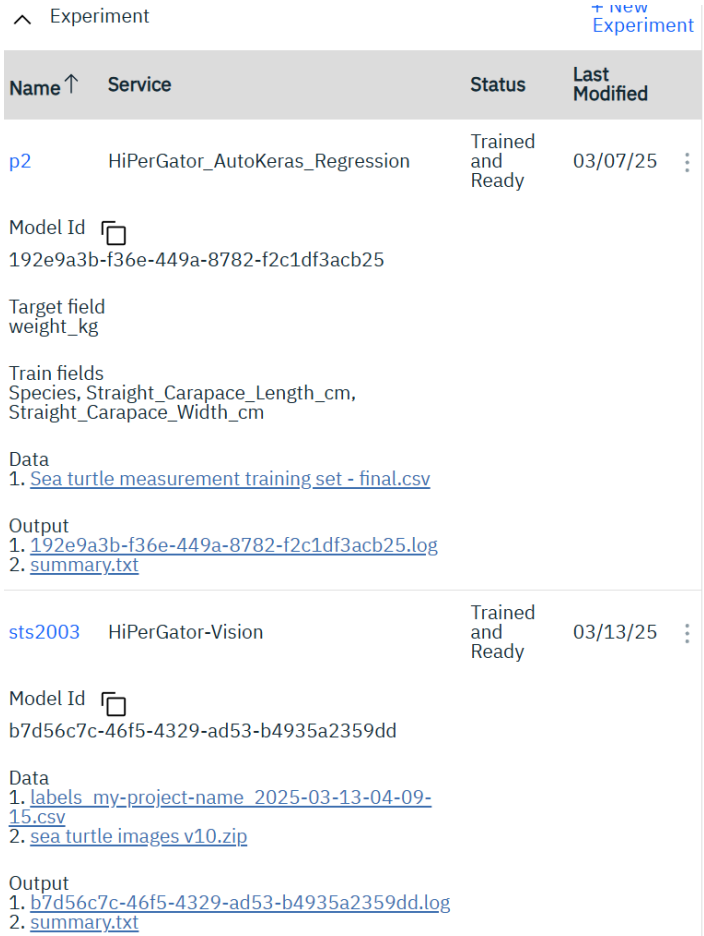
The details of the model are below, and it shows the different techniques used and outputs given. One detail I find important is the number of parameters. In the dense model, 1056 parameters are used.

```
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 3)]               0
_________________________________________________________________
multi_category_encoding (Mul (None, 3)                 0
_________________________________________________________________
normalization (Normalization (None, 3)                 7
_________________________________________________________________
dense (Dense)                (None, 32)                128
_________________________________________________________________
re_lu (ReLU)                 (None, 32)                0
_________________________________________________________________
dense_1 (Dense)              (None, 32)                1056
_________________________________________________________________
re_lu_1 (ReLU)               (None, 32)                0
_________________________________________________________________
dense_2 (Dense)              (None, 32)                1056
_________________________________________________________________
re_lu_2 (ReLU)               (None, 32)                0
_________________________________________________________________
regression_head_1 (Dense)    (None, 1)                 33
=================================================================
Total params: 2,280
Trainable params: 2,273
Non-trainable params: 7
_________________________________________________________________
```

# Part 3 - Image Recognition

## Process

In part 3, I trained an image recognition model to recognize different species of sea turtles. The process involved labeling images of sea turtles, uploading the annotated images to Cammy, and training the AI model on the image data. The result was a model that could take pictures of turtles of one of the 4 species and predict which species is shown in the image.



Experiment

+ New Experiment

| Name | Service | Status | Last Modified |
|---|---|---|---|
| p2 | HiPerGator_AutoKeras_Regression | Trained and Ready | 03/07/25 |

Model Id
192e9a3b-f36e-449a-8782-f2c1df3acb25

Target field
weight_kg

Train fields
Species, Straight_Carapace_Length_cm, Straight_Carapace_Width_cm

Data
1. Sea turtle measurement training set - final.csv

Output
1. 192e9a3b-f36e-449a-8782-f2c1df3acb25.log
2. summary.txt

| | | | |
|---|---|---|---|
| sts2003 | HiPerGator-Vision | Trained and Ready | 03/13/25 |

Model Id
b7d56c7c-46f5-4329-ad53-b4935a2359dd

Data
1. labels_my-project-name_2025-03-13-04-09-15.csv
2. sea turtle images v10.zip

Output
1. b7d56c7c-46f5-4329-ad53-b4935a2359dd.log
2. summary.txt

Services Used to train model (HiPerGator)

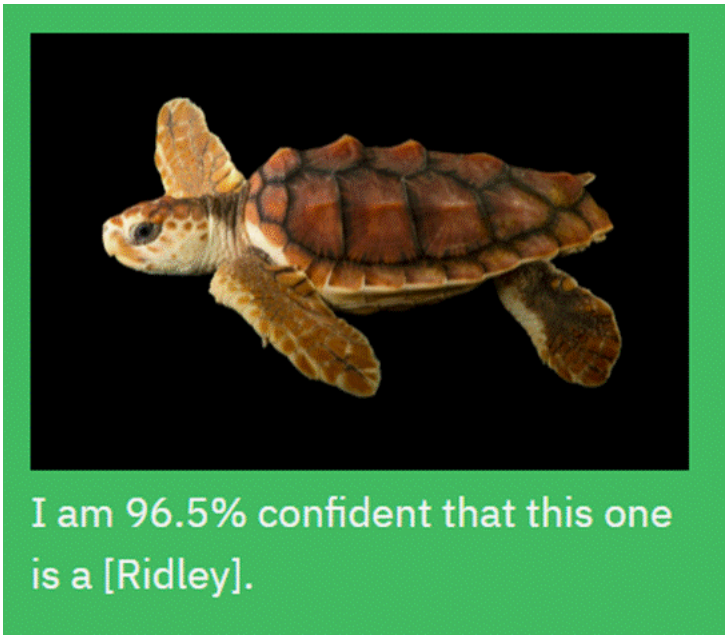
```
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 180, 180, 3)]     0
_________________________________________________________________
cast_to_float32 (CastToFloat (None, 180, 180, 3)       0
_________________________________________________________________
normalization (Normalization (None, 180, 180, 3)       7
_________________________________________________________________
conv2d (Conv2D)              (None, 178, 178, 32)      896
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 176, 176, 64)      18496
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 88, 88, 64)        0
_________________________________________________________________
dropout (Dropout)            (None, 88, 88, 64)        0
_________________________________________________________________
flatten (Flatten)            (None, 495616)            0
_________________________________________________________________
dropout_1 (Dropout)          (None, 495616)            0
_________________________________________________________________
dense (Dense)                (None, 4)                 1982468
_________________________________________________________________
classification_head_1 (Softm (None, 4)                 0
=================================================================
Total params: 2,001,867
Trainable params: 2,001,860
Non-trainable params: 7
_________________________________________________________________
```

Model training summary

## How can we improve ML models?

To improve machine learning models, a wide variety of data is required. I trained the model from the training set and tested it using the testing set, and it got some of the pictures wrong; however, the accuracy and confidence for predictions were higher for the testing set than for images from the internet. With a wider variety and number of predictions, it would be more accurate and confident in its correct predictions and less confident in its incorrect predictions.


The model correctly identifies a turtle from the testing set as Green.




The model incorrectly and confidently identifies an image from the Internet.

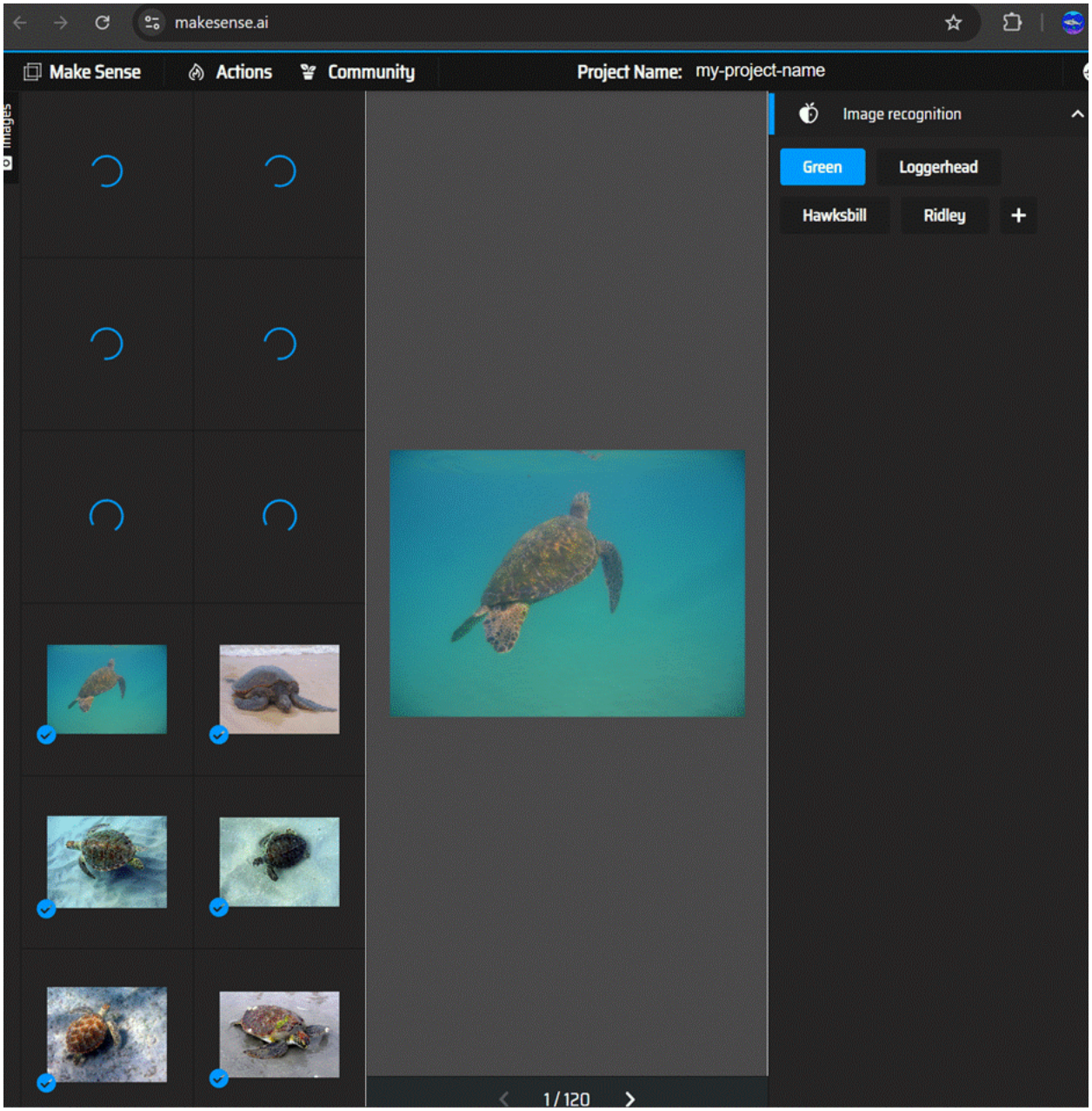
## Training time

The cogbot learned in approximately 20 minutes from the data. This could be sped up with more processing power, and it took quite a long time to train from only 120 images. Training AI models is computationally expensive and takes a long time. It is not possible to train an AI model instantly.

## Training Issues

One thing that could be causing the AI model to fail to recognize certain images is that it needs a lot more data. Images have millions of pixels and features that could impact the species. A data set of size 120 is not enough for highly accurate predictions. It also could have some built-in bias from the images that were fed to the model.

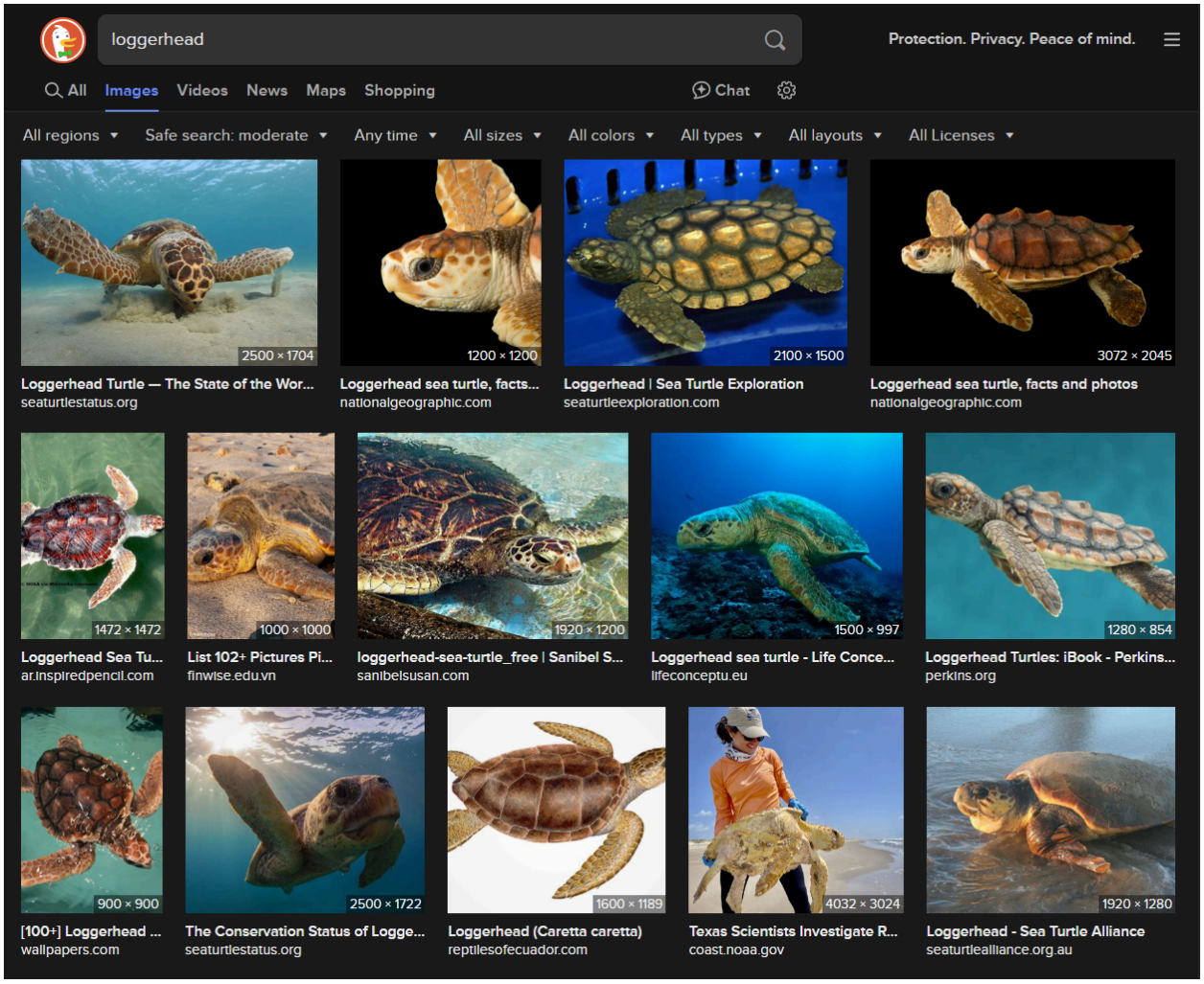
A problem I ran into during my project was some of the images in the provided data set would not load during annotating. I was still able to label the images because of the file name, but it could impact how the model interprets the data.




Images not loading problem

## Image Sourcing

A way to find images is to use a search engine like Google and input the annotations manually. This can be time-consuming, and a better resource for datasets is Kaggle, which has all kinds of tables and different types of data sets.

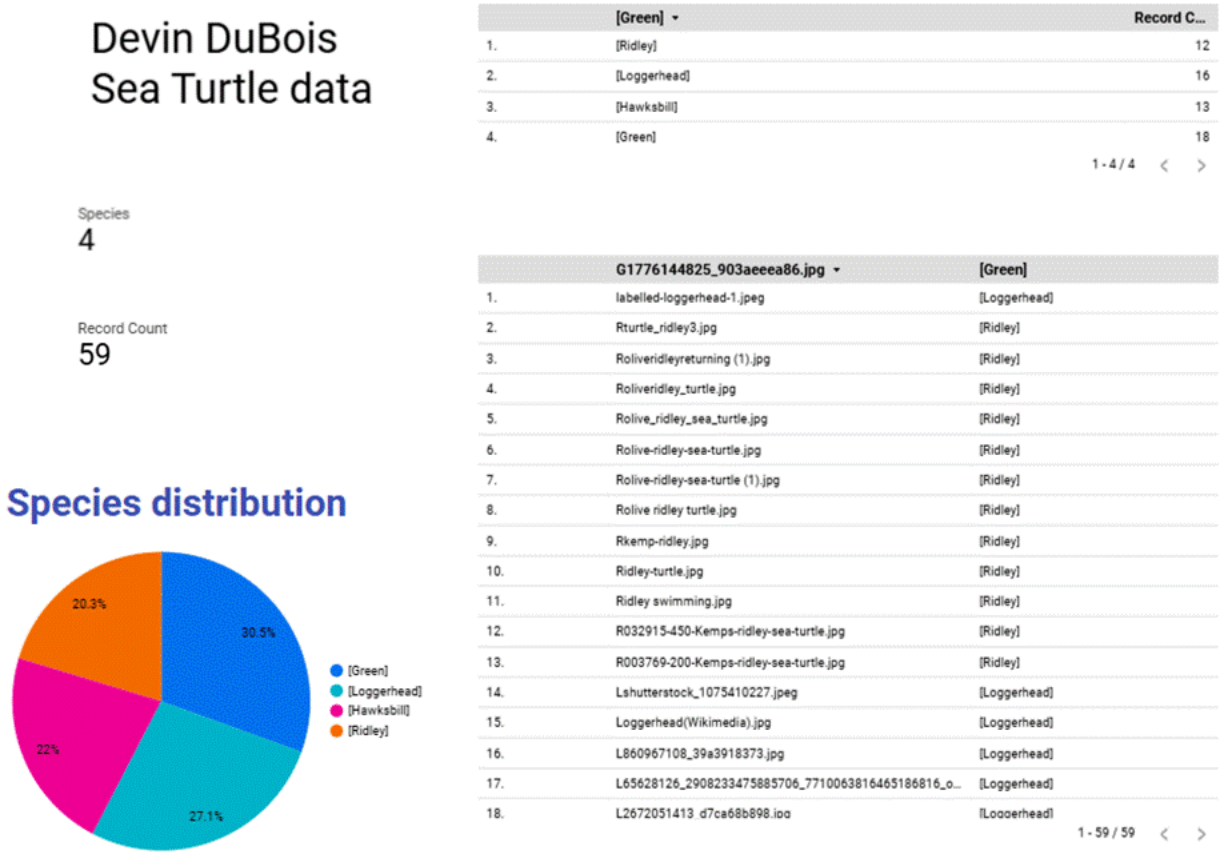



Search Result example

## Bias and Ethical Problems

Image recognition could be a concern for our society if it is in the hands of authoritarian governments. One example that comes to mind is China using facial recognition on street cameras to identify people to adjust their social credit score.

Image recognition could be biased because of the data it was fed. For example, if all images of a Ridley Sea turtle have a dark background, the model would associate a Ridley turtle with a dark background. This issue can be resolved by inputting a wide variety of data. In this example, putting different types of Ridley turtles with different colors and background colors will help the model better identify pictures.




Visual overview of the dataset

# Conclusion

This was a great experience for learning about AI and how different ML techniques are used to make predictions about data. It did a good job of showing how inputting more data increased the accuracy of subsequent questions. Google Studio is a great tool for data visualization as it simplifies the process of showing the summary of different aspects of the data. The biggest issue was for image recognition where the accuracy was very low, and a recommendation I have is to increase the size of the dataset for comparison, for example to 5000 images, and compare performance. Overall, teaching at CogUniversity was an interesting experience and a great tool for learning about AI.

# References

https://cu2.uf-cap1.cogability.net/#/en-us/

https://cu2.uf-cap1.cogability.net/#/en-us/classrooms/course/STB1002_EEL3872/Spring%202025

https://cu2.uf-cap1.cogability.net/#/en-us/classrooms/visionClassroom/Sea%20Turtle%20Weight%20Prediction%20-%20TWP2002_EEL3872/TWP2002_EEL3872/Spring%202025

https://cu2.uf-cap1.cogability.net/#/en-us/classrooms/visionClassroom/Sea%20Turtle%20Species%20Identification%20-%20STS2003_EEL3872/STS2003_EEL3872/Spring%202025

# Appendix

None is needed for this report.