SUPPLEMENTARY MATERIAL

SnakeChat: a conversational-AI based app for snake classification

Jorge Guerra Pires [1]

[1]Founder at IdeaCoding Lab / JovemPesquisador.com, Brazil

*jorgeguerrapires@yahoo.com.br

Abstract

This document provides details on the Teachable Machine models.

Keywords: biology; bioinformatics; conversational artificial intelligence; Snakes; openAI; chatGPT; chatbots.

1 Supplementary Material

On this section, I have organized materials that I believe can be useful for the reader, but does not effect the understanding of the main proposal. The reader may consult this material on a more detailed reading of the main paper. I have done my best to include all the main extra materials, avoiding unnecessary information. For more details, the reader is kindly invited to get in touch with me.

On Section 1.1, I present the models under the hood used on the second version of the chatbot. Those are the image identification models, the computer vision models that the chatbot calls for making a prediction of a snake using an image. See Pires and Dias Braga (2023) for a more detailed account of the methods used on those models. I have applied the same approach herein.

For real conversations and discussions on the chatbot behaviors, see the additional SM.

1.1 Details on the model training using Teachable Machine

On this section, we have the numerical results for training the models under the hood, that the chatbot can call and use when needed. All the models were trained using Teachable Machine (TM).

I have trained several models. Three of them are specific: a model for true coral snakes (Section 1.1.3), a model for false coral snakes (Section 1.1.4) and a model for the *bothrops* family (Section 1.1.5). The general models are: a model for separating between the true and false coral snakes (Section 1.1.1) and a general model which includes all the species from the previous models Section 1.1.2.

What I have learnt, also comparing with the model from INaturalist and MobileNet: the more general is the model, the higher the chances of misclassifications. As we go general, adding new species, the higher the changes of "bizarre" classifications, sort of "hallucination". For instance: a snake be confused with a duck, or a boat. A grass be confused with a crocodile. When we need precision, high accuracy, a diagonal confusion matrix, smaller models are better.

All the species of snakes were collected from *Quadrilátero Ferrífero*. The *Quadrilátero Ferrífero* is a region located in the central-southern part of the state of Minas Gerais, Brazil. It is an area of approximately 7,000 square kilometers and is home to several important cities such as Sabará, Rio Piracicaba, Congonhas, Casa Branca, Itaúna, Itabira, Nova Lima, Santa Bárbara, Mariana, and Ouro Preto.

The Quadrilátero Ferrífero is a region of great economic importance to Brazil, as it is the country's main producer of iron ore [1]. However, mining activity in the region has been associated with environmental problems such as loss of biodiversity and pollution of soil and groundwater [2].

The following species were added to the models. The first column is the name in the model, the second column is the complete name (genus + species), given to the language model. Initially, it was given just the shortened name, but the language model would sometimes guess wrongly the complete name. That would trigger additional information added by the language model to be wrong. Those additional information they add can be interesting to read. They can be very informative.

Model for fake vs. true coral snake:

i. "A. assimilis": "Apostolepis assimilis",
ii. "E. aesculapii": "Erythrolamprus aesculapii",
iii. "O. rhombifer": "Oxyrhopus rhombifer",
iv. "M. corallinus": "Micrurus corallinus",

Accuracy per class

CLASS	ACCURACY	# SAMPLES
A. assimilis	0.92	12
E. aesculapii	0.69	13
O. rhombifer	0.64	11
M. corallinus	0.73	11
M. frontalis	0.67	12
M. carvalhoi	0.64	11

Figure 1: Accuracy per class for the false vs. true coral snake model. Values range from 0-1, which is the numerical equivalent of 0%-100%.

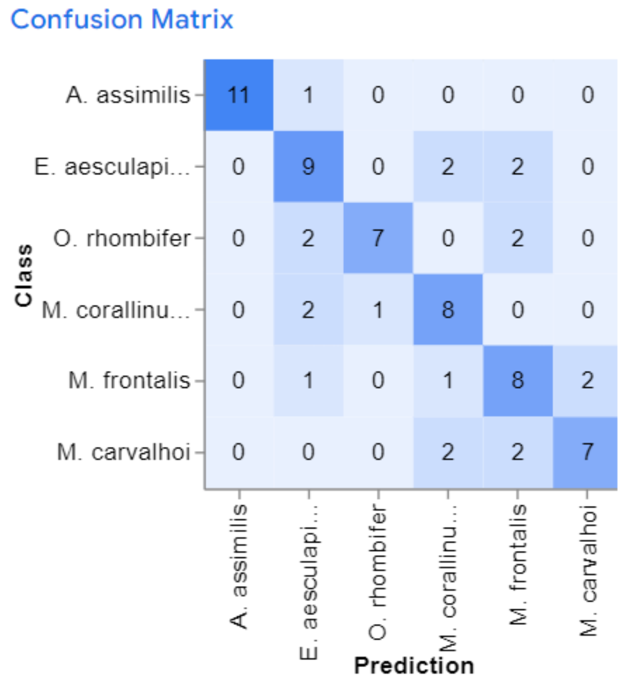

Figure 2: Confusion matrix for the true vs. fake coral snake model.

v. "M. frontalis" : "Micrurus frontalis",
vi. "M. carvalhoi": "Micrurus carvalhoi",

Bothrops snakes:

i. "B. neuwiedi": "Bothrops neuwiedi",
ii. "B. jararaca": "Bothrops jararaca",
iii. "B. alternatus": "Bothrops alternatus"

All the other models are grouping of those species.

1.1.1 Coral snakes vs. false coral snakes

Fig. 1 illustrates the accuracy per class for the model that tries to separate between true and false coral snake.

This model is hard to converge, it was tried several configurations. One of them was to lower the learning rate to 0.000001 (the default is 0.001). It took about 30 minutes to converge, and almost 5.000 iterations (generally, it converges in less than 5 minutes, in about 50 iterations). The final result was no better than the one reported herein. It generally happens when the model is complex, which is our case. The simulation was done several times. The model that had a good accuracy per class was selected as final. The accuracy per class is at least 50%, 16% is a measure of random prediction. Thus, even though our model is not as high as we would like to, it is still better than someone randomly choosing a snake species from a list. Therefore, our model is better than random guessing.

Fig. 2 illustrates the confusion matrix for the true vs. false coral snake model. The model has several misclassifications. Those misclassifications reflect the difficulty to have a big model for both true and false coral snakes: false coral snakes try to mimic the true ones. Thus, this "confusion" is nature's way to fool predators. The table is organized on sections: it is possible to see visually that the sub-matrix around the main diagonal on the lower-right corner is where are most of the misclassifications. This is the true coral snake section: the model just for those snakes does much better. In fact, different from the false coral snakes, the true coral snakes are very alike, they belong to the same genus (*Micrurus*).

Fig. 3 is the overall accuracy for the model. This is generally the best accuracy that was achieved after several attempts, one including a very slow learning rate, which took about 30 minutes to converge. The result was even worse for this long-lasting simulation.

Fig. 4 is the loss function for the true vs. false coral snakes. The validation curve (orange) is upwards, which means that it did not generalized properly. Several attempts were done, one with a very low learning rate, this behavior is predominant. This means that the model is complex: it is not simple to separate false from true coral snakes. Another possibility is that we need more images. Now I have used about 70 images per class, and the images were collected from INaturalist and Google Image. This is something that may be tried in the future, but based on the simulations, it does not seem to be the case. I have started with about 10 images per class, and added those extra images for the current paper. The behavior is the same. See where I have documented the model with a small number of images Pires and Dias Braga (2023).

1.1.2 General model

This model has all the species of the sub-models. This a prototype towards a big model for snake classification. The strong side of sub-models is that they do very good at the species they were trained to classify, but they will also classify anything that is entered on the model, with a high

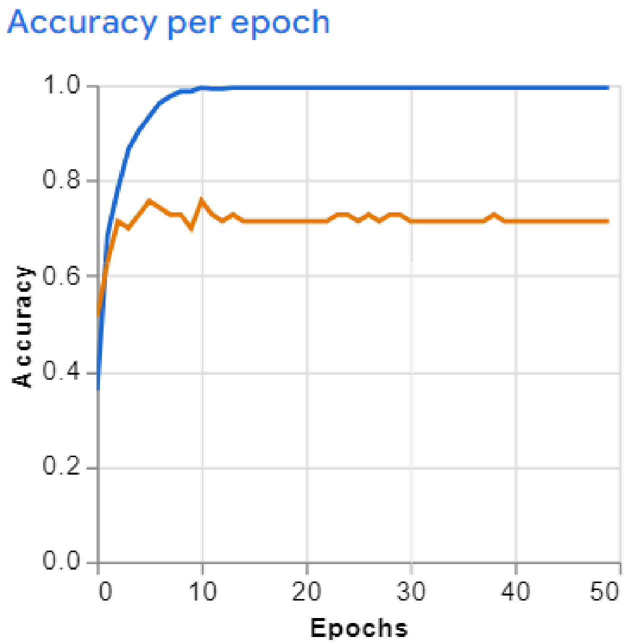

Figure 3: Overall accuracy for the true vs. false coral snake model.

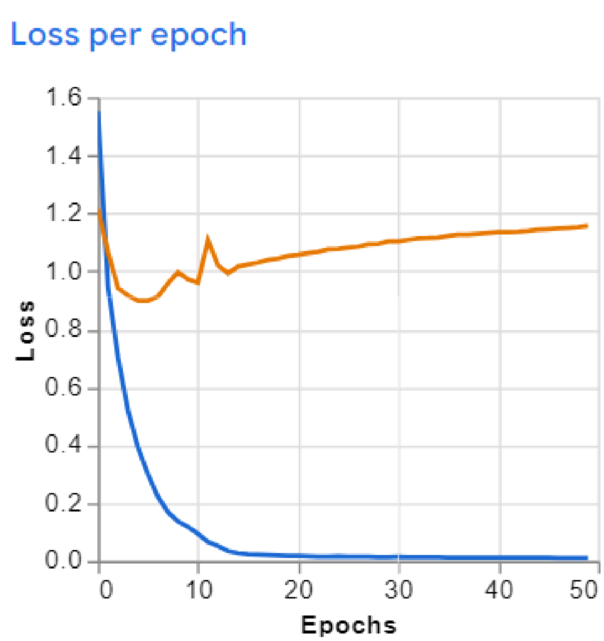

Figure 4: Loss function for the true vs. false coral snake model.

Accuracy per class

CLASS	ACCURACY	# SAMPLES
O. rhombifer	0.55	11
A. assimilis	0.83	12
E. aesculapii	0.69	13
M. carvalhoi	0.73	11
M. corallinus	0.64	11
M. frontalis	0.50	12
B. alternatus	0.91	11
B. jararaca	0.92	12
B. neuwiedi	0.64	11

Figure 5: Accuracy per class for the general model.

probability in some cases. A model will fit any image on the best they have, which means also making a meaningless classification. If you enter a *Bothrops* snake on a *Micrurus*, it will classify, yielding even a high probability. Putting it directly: computer vision models are good at what they were trained upon, but bad at what they were not trained upon. They see what you trained them to see, but see what you trained them to see where it does not exist (a sort of hallucination). Therefore, the right model should always be used.

Fig. 5 is the accuracy per class for the general model. As it is possible to see, the accuracy per class changes from class to class. The model was trained several times, and an accuracy per class that was at least 50% was chosen. It has a difficult convergence: most of the time, it would not converge, and the result I report was amongst the best that appeared during those simulations. I stopped the simulation when I judge it was okay, based on previous results. if left running, chances are high the validation loss function would go upward.

Fig. 6 is the confusion matrix for the general model. It is possible to see that the misclassifications are in blocks. It happens because the species are organized in blocks, the same blocks used for the sub-models. It helps on the visualization. Thus, the model is misclassifying where the snakes are very similar, what I already expected.

Fig. 7 illustrates the general accuracy for the general model. This is the best possible for this big model. The classification is hard as I could see from the simulations. The simulations were stopped since they tend to go for

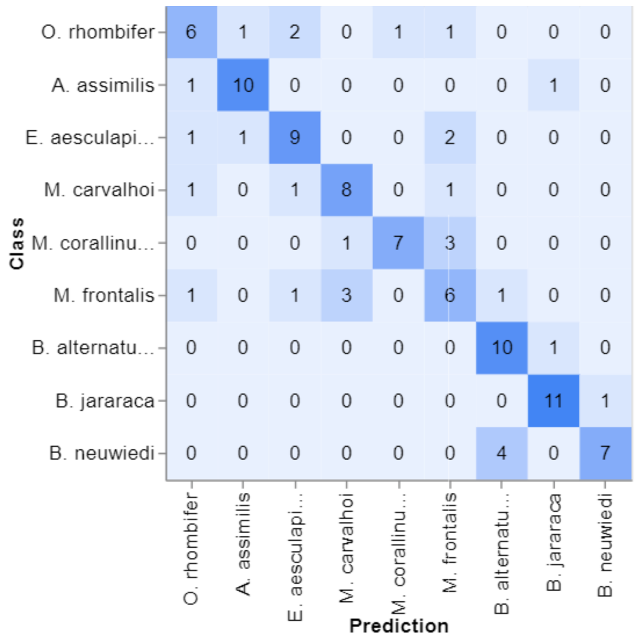

Figure 6: Confusion matrix for the general model.

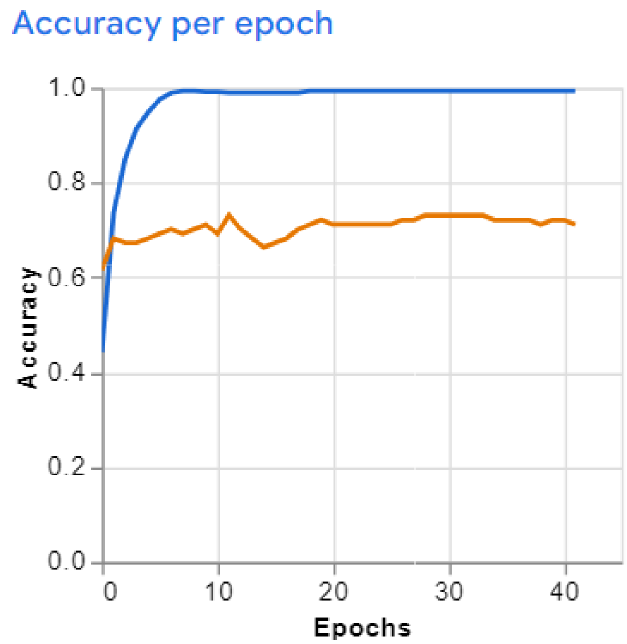

Figure 7: General accuracy for the general model.

worse results, just the loss function for the validation curve. The training curve always converged, and stayed high. It is important to avoid overfitting, at least try to avoid this undesirable result.

Fig. 8 is the loss function for the general model. The loss function for the validation tends to go up. The simulation was stopped since this is generally the best result it is possible to get.

1.1.3 *Micrurus models*

Even though fake coral snakes belong to different species, the true ones belong to just one genus: *Micrurus*.

For selecting the model, I have trained several times using TM. I have selected the models with overall accuracy of 80% for the validation curve, the training curve most the time arrived to almost 100% of accuracy. From these models, I have selected the one that had an well-distributed accuracy per class, at least 80% per class. I have also taken a look on the confusion matrix: I looked for a homogeneous misclassification pattern. A model that is very good on a species, and very bad on others was not considered a good model, even when the overall accuracy was good. The rationale is not having an unbalanced model: a model that performs very well on a class, and very poorly on another. The threshold for random guess is 34%: this is the chance of getting the species right just by guessing randomly.

I have made a balanced number of samples per species, for avoiding an unbalanced model, a biased model towards a species. This same care was taken with all the other models. When a unbalanced model is trained, it will perform very well on that species. The model tends towards the species with more images. And our goal is a balanced model.

Image per class, the same number was used for the other models: about 70 images per class:

Figure 8: Loss function for the general model. Training in blue, and validation in orange

Accuracy per class

CLASS	ACCURACY	# SAMPLES
Micrurus corallinu...	0.82	11
Micrurus frontalis	0.92	12
Micrurus carvalhoi	0.91	11

Figure 9: Accuracy per class for the *Micrurus* model

i. *Micrurus corallinus* - 73 images;
ii. *Micrurus frontalis* - 75 images;
iii. *Micrurus carvalhoi* - 71 images;

All the images were selected from *INaturalist*. Five images were selected for testing the model, those images were not presented to the model during training. I have also used validation images, which is standard on TM: they handle these separation during training. Thus, the validation process made sure the model was well-trained: the five images were just for a final test, an extra test.

Fig. 9 illustrates the accuracy per class for the *Micrurus* model. As it was set for the model selection: the accuracy per class is equally distributed, at least 80% per class.

Fig. 10 illustrates the confusion matrix for the *Micrurus* model. One thing that we should keep in mind: those snakes are very alike, and it is a curious result that we can get this level of accuracy. Talking to a biologist informally, he found it surprising the good result between *Micrurus carvalhoi* and *Micrurus corallinus*: they are very alike. In fact, I have explored a feature from TM that allows us to disable a class during training. When all those snakes were put as pairs, a binary model, the accuracy were almost 100% , and the confusion matrix was either diagonal, or almost diagonal.

Fig. 11 illustrates the overall accuracy for the *Micrurus* model. The training was stopped after about 80% of accuracy for the validation curve (orange) was achieved. The simulation was repeated several times, and this is generally the best the model can do.

Fig. 12 illustrates the loss function for the *Micrurus* model.

The final model is available at *Micrurus* model. This is the same link used to upload the model on Angular, on the chatbot. It is possible to use this model on your own research, should you judge useful. The model is ready to use, a pretrained model.

1.1.4 *Fake coral snakes*

Fig. 13 illustrates the loss function for the fake coral snake model. The training process converged. When we compare this model with the general models (Section 1.1.1 and Section 1.1.2), it is clear that this specialized model is better both at converging and at accuracy per class. Even though

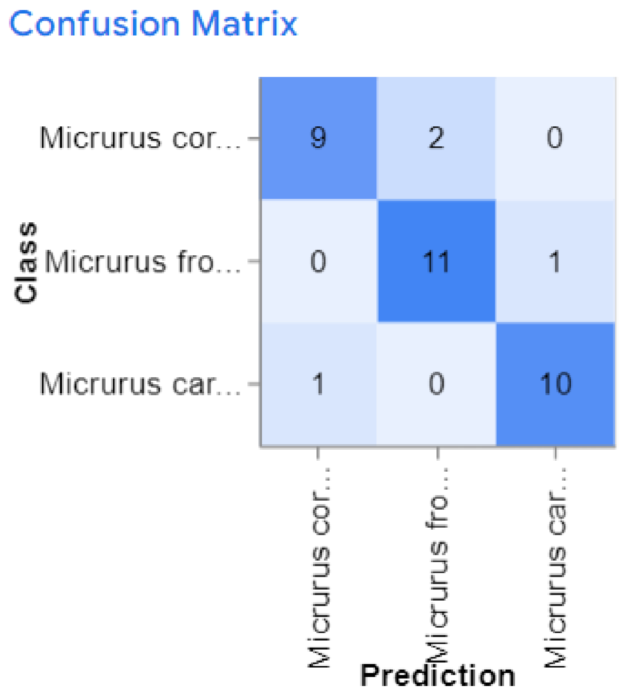

Figure 10: Confusion matrix for the *Micrurus* model.

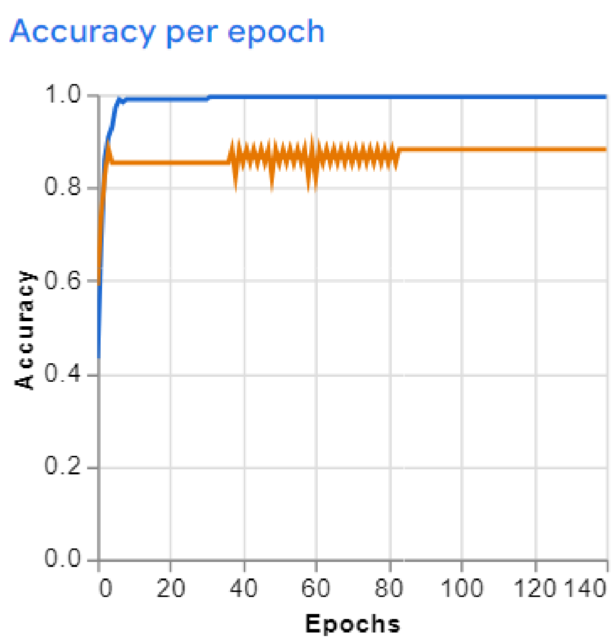

Figure 11: Overall accuracy for the *Micrurus* model.

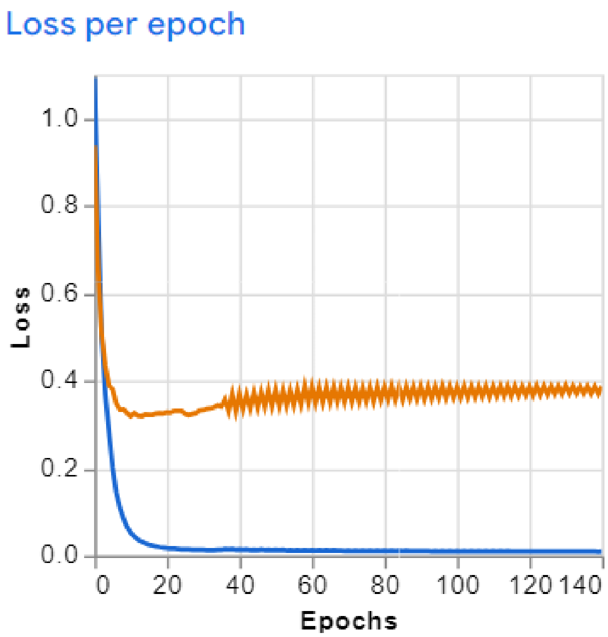

Figure 12: Loss function for the *Micrurus* model. Training in blue, and validation in orange.

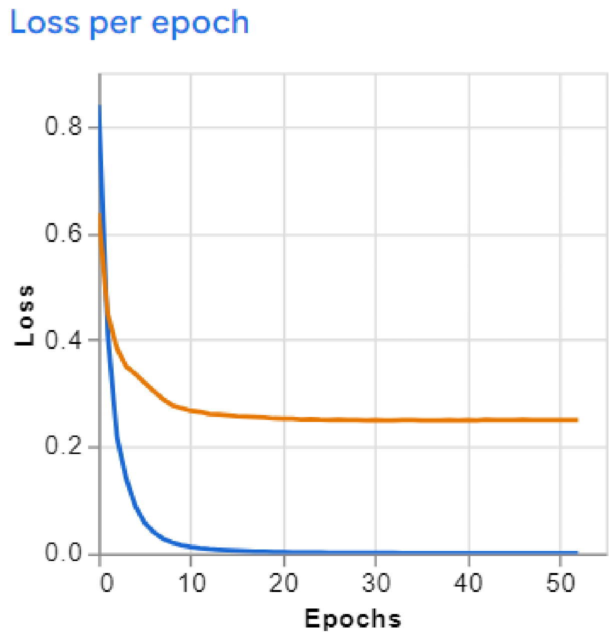

Figure 13: Loss function for the fake coral snake. Training loss in blue, and validation in orange.

the false coral snakes belong to different species, they are easy to separate when they are placed at a "group" (this is not a scientific group *per se*).

Fig. 14 illustrates the overall accuracy for the fake coral snake model. The overall accuracy is about 90% for both training and validation.

Fig. 15 illustrates the confusion matrix for the model. The matrix shows that the model have learnt very well, with small misclassifications. See that most of the misclassifications are between *Oxyrhopus rhombifer* and *Erythrolamprus aesculapii* (e.g., Fig. 16). If you try to enter that image in our model, it will classify as 100%, that is because this image was added to training dataset. Initially, I have removed this image thinking it was a mistake, until a biologist callled my attention that it was actually a *Erythrolamprus aesculapii*, not a *Oxyrhopus rhombifer*. I have done a Google Image reverse, they also make the mistake. BARD (Google) curiously got it right, it does not recognize *Oxyrhopus rhombifer* at all. It is possible to find all the BARD conversation here.

Fig. 17 illustrates the accuracy per class. The model was trained several times, and the chosen model was the one that kept an accuracy per class well-balance. it is at least 80% per class.

The final model is here for usage.

1.1.5 Bothrops model

On this section, the numerical results for training the model for the *Bothrops* genus are presented.

For this case, it was necessary to try several times for getting a good result, and the result that was achieved is a very good result. It happens generally when the classification is not easy: the snakes are very similar, like

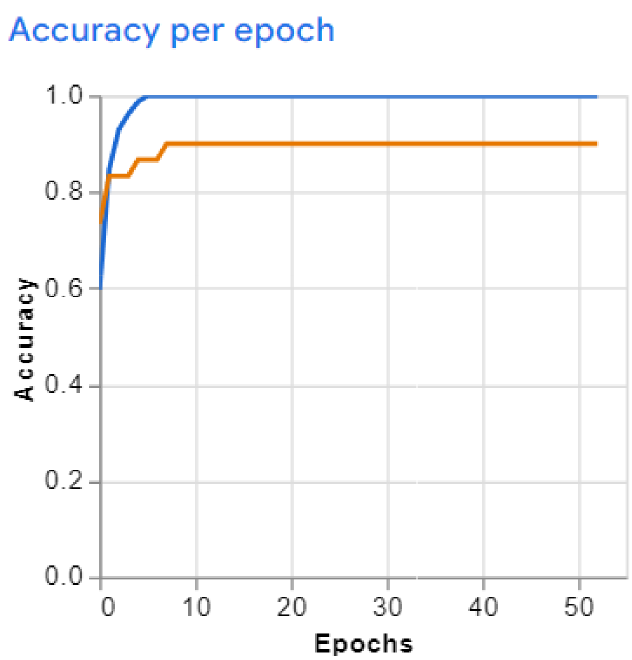

Figure 14: Overall accuracy for the fake coral snake model. Training loss in blue, and validation in orange.

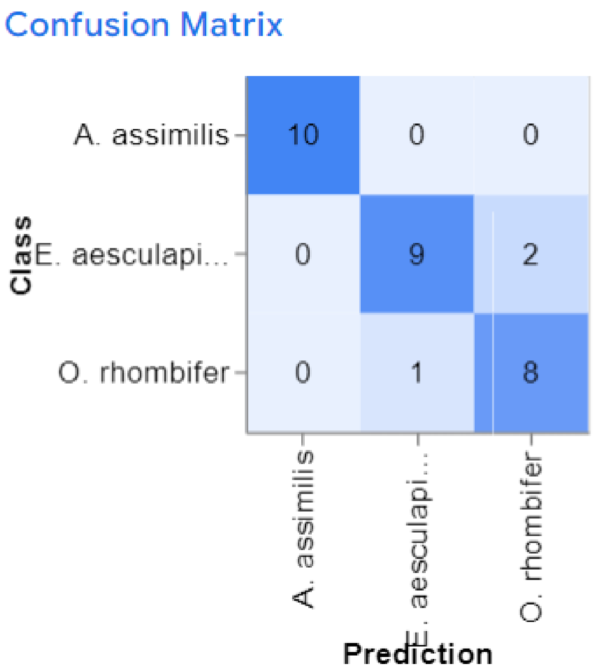

Figure 15: Confusion matrix for the fake coral snake model.

Figure 16: *Erythrolamprus aesculapii* that can be confused with *Oxyrhopus rhombifer*. Source: Google Image. This image may be protected by copyright.

Accuracy per class

CLASS	ACCURACY	# SAMPLES
A. assimilis	1.00	10
E. aesculapii	0.82	11
O. rhombifer	0.89	9

Figure 17: Accuracy per class on the fake coral snake

separating fake from true coral snakes. TM is using back-propagation, a local search method. Therefore, since the neural network weights are randomly started, it is natural to start on "a bad place". After training several times, it is possible to achieve a very good result. Also, some of the snakes have too much variations in colors for the same snake species.

It is possible to see from images that this family may be very rich in coloration and patterns. After talking to a biologist, it became clear to me, for instance, that *bothrops alternatus* may have several variations in colors and patterns. It is possible to note that when preparing the images for training.

Fig. 18 illustrates that the model converged, both the training and validation curves. It converged to a low value, and both remained on this low value. This is one of the best results that could be expected from such a training process.

Fig. 19 illustrates that both the training and validation achieved an accuracy of about 100%. Since both the training and accuracy arrived to this value, it is unlikely that we had an overfitting. For achieving this result, it was necessary to train several times. Each time we trained, the model would be randomly initialized.

Fig. 20 illustrates the confusion matrix. There were no misclassifications. This was already expected since the accuracy per class is 100%.

The final model is available here, this is the same link used to upload the model locally on Angular, that the chatbot uses to make the classification.

References

Pires, J. G. and Dias Braga, L. H. (2023). Snakeface: a transfer learning based app for snake classification, *Revista Brasileira de Computação Aplicada* **15**(3): 80–95. **URL:** *https://seer.upf.br/index.php/rbca/article/view/15028*

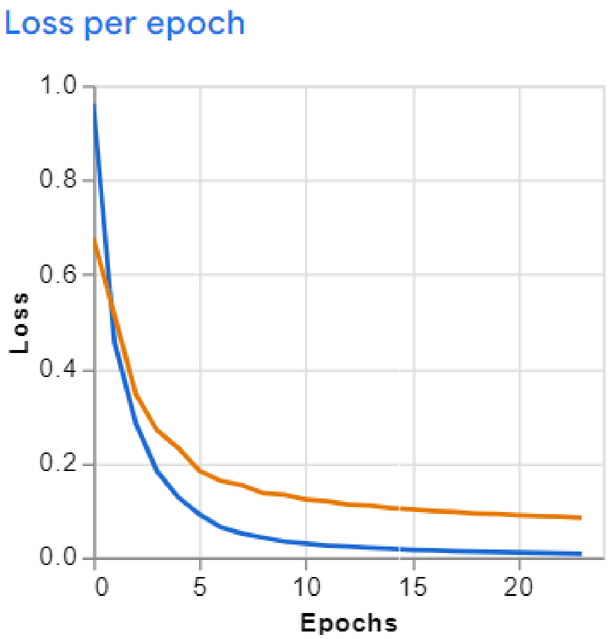

Figure 18: Loss function for the Bothrops model. Legend: training in blue, and validation in orange.

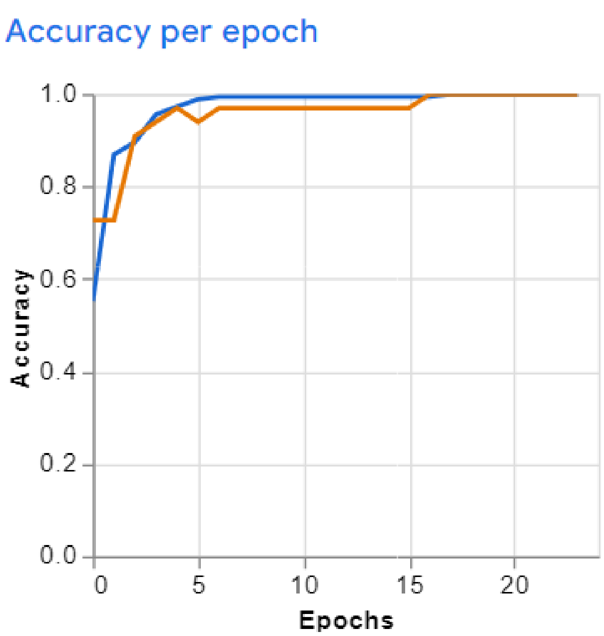

Figure 19: Accuracy for Bothrops model. Legend: training in blue, and validation in orange.

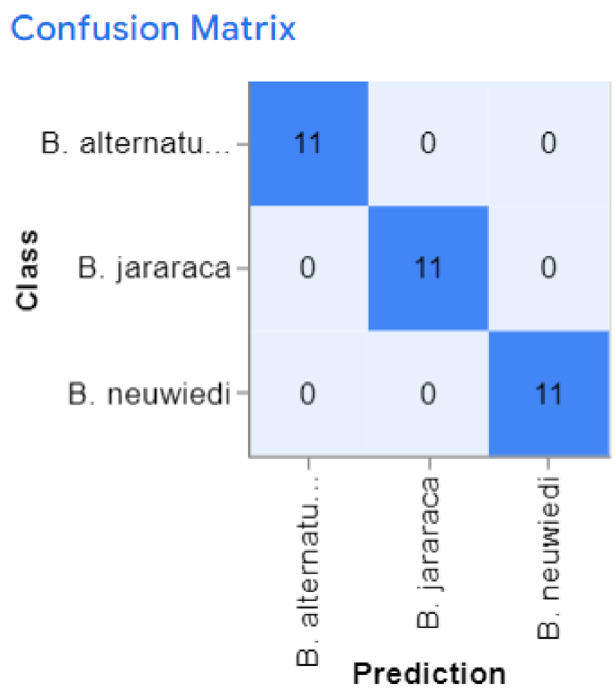

Figure 20: Confusion matrix for Bothrops model.