FIRE: THE FIRST-YEAR INNOVATION & RESEARCH EXPERIENCE
CLOUD COMPUTING

Improving Automated Polar Low Classification

Shraddha Patre, Riya Kanani, Katie Fernon, Ryan Birmingham

Introduction

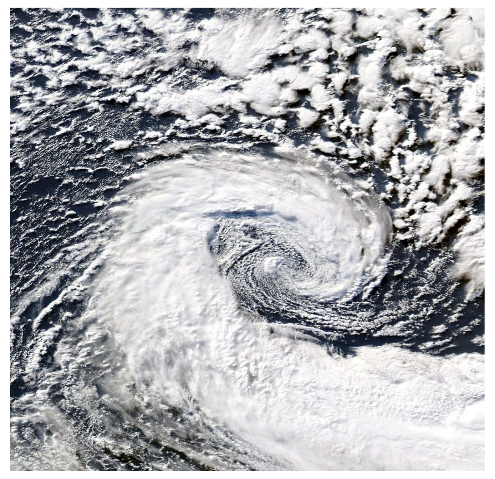

Figure 1. Polar Low

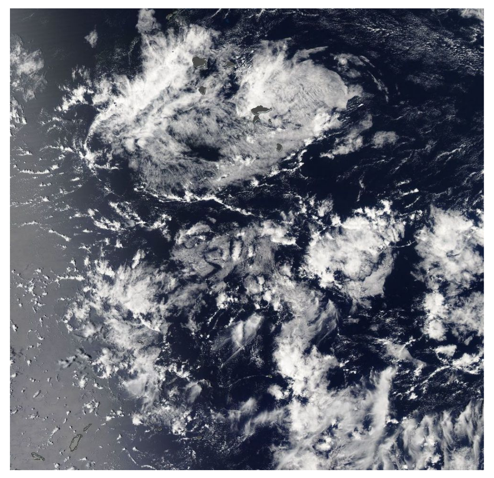

Figure 2. Non- Polar Low

Background: Polar lows are fast-forming storms accompanied by strong winds and hail or snow. They have destructive capabilities that can harm communities if they're not prepared for the storm.[1]

Motivation: We are motivated to achieve accurate polar low classification and further image identification research. Previously, our team used Google Teachable Machine combined with a decision tree to approach this problem. We aim to improve our previous approach by replacing the usage of Google Teachable Machine with a more effective CNN model.

Methods

Tools:

- AlexNet: A pretrained Convolutional Neural Network that we re-trained on our images to identify polar lows and non-polar lows.[2]
- PyTorch: An open-source deep learning framework we used to re-train the AlexNet model
- Scikit-learn: A software machine learning library for Python - used for building decision tree classifiers.
- Decision Tree: A decision tree is a supervised learning algorithm that can be used for image classification

Data:

- NASA Worldview: A data set of satellite images that we used to capture images of polar lows and non-polar lows that were 1000 x 1000 pixels and 1km.
- Pickle Files: Filtered data from previous FIRE semesters containing images of polar lows and non-polar lows, along with metadata such as longitude and latitude data.

Results

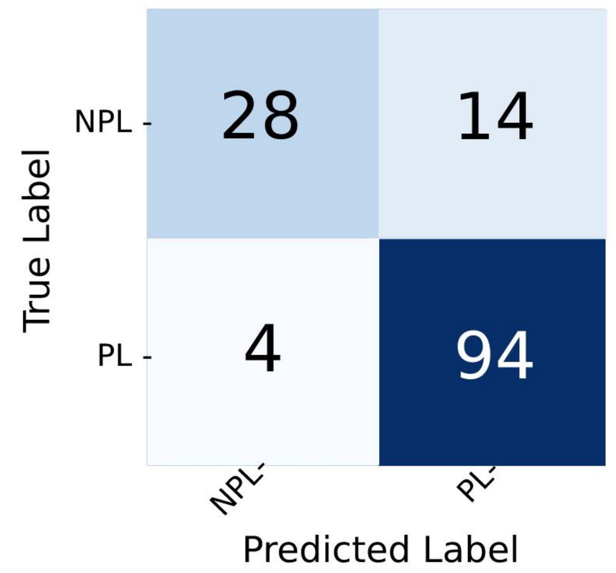

Figure 3. This is the confusion matrix for the AlexNet model.

Accuracy of our AlexNet model at 100 epochs (Fig. 3)

- Polar Low (PL) Accuracy: 95.92%
- Non-Polar Low (NPL) Accuracy: 66.67%
- Overall Accuracy: 81.29%

In image classification, an epoch is the number of times the learning algorithm goes through the dataset.

This data shows us that the AlexNet model was able to classify polar low images better than non-polar low images after 100 epochs.

Initially, we chose to train our model with 100 epochs to see the variation in training and testing loss/accuracy (see Fig. 4 and 5) with an increasing number of epochs. Loss is the summation of errors that our model produces. According to Fig. 4, as the epochs increased, the test loss increased, but the training loss decreased, signifying overfitting.

As observed in Fig. 5, as the epochs increased, the training accuracy drastically increased in the beginning but had minor variations after around 5 epochs. The testing accuracy decreased overall between 0 and 100 epochs, however, it varies significantly between each epoch. This signifies that after 5 epochs, the model begins overfitting. Therefore, for this model, 5 epochs is the most optimal to avoid overfitting issues.

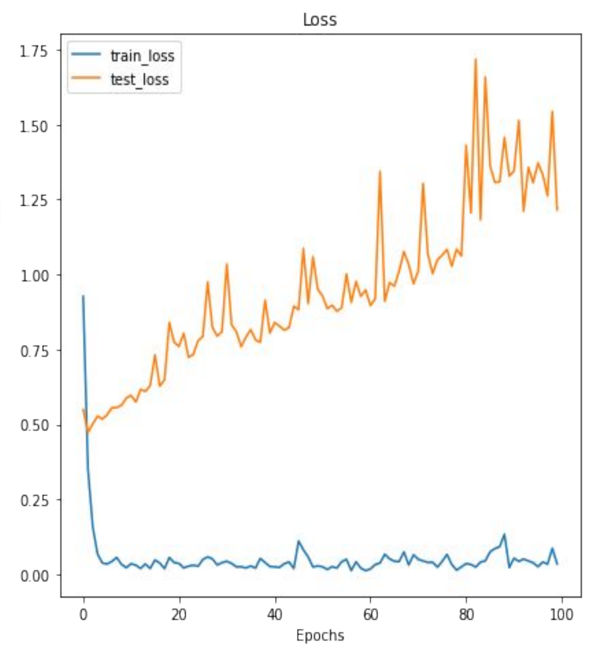

Figure 4. Epochs vs Loss: This graph measures the loss of our model depending on the number of epochs

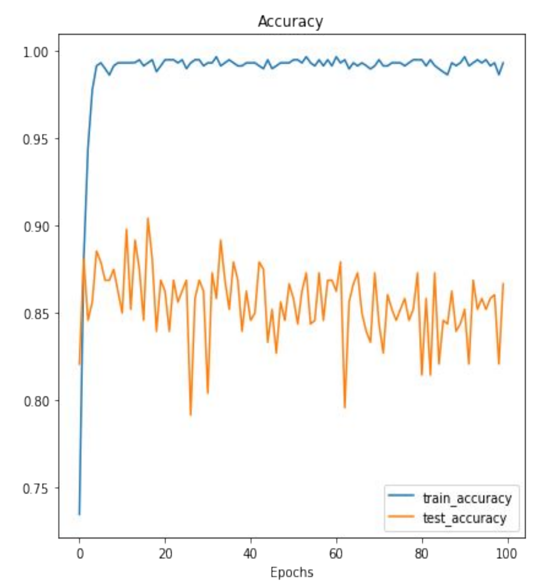

Figure 5. Epochs vs Accuracy: This graph measures the training and testing accuracy of our model depending on the number of epochs

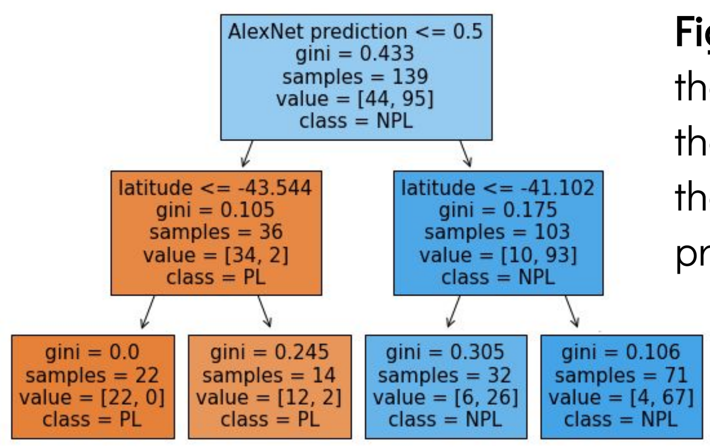

Figure 6a: This is the decision tree that incorporates the AlexNet predictions

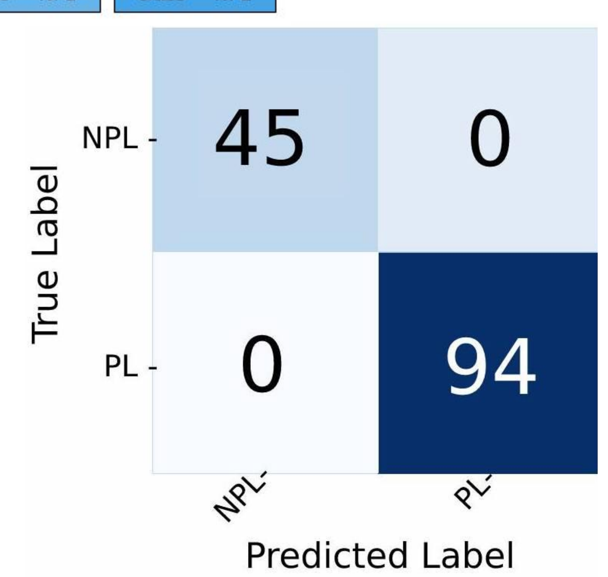

Figure 6b: This is the confusion matrix after implementing the decision tree

Fig. 6a represents the decision tree we created last semester. After some trial and error, we found that our decision tree has the most accuracy when it has three layers.

Fig. 6b shows the accuracy of our decision tree when combined with the AlexNet model above. This combination was achieved by feeding the raw predictions from AlexNet at 5 epochs into the decision tree.

- Polar Low Accuracy: 100%
- Non-Polar Low Accuracy: 100%
- **Overall Accuracy: 100%**

Discussion

Last year, our team's model had an accuracy rate of 73.5%. This semester, in FIRE298, to improve the accuracy of our model, we utilized a pre-trained CNN model, AlexNet, to initially classify a collection of 139 images after being re-trained with 556 images. The accuracy achieved by this model was 81.29% - an 8% increase from the previous model. Similar to last year, our team sought to use a decision tree to increase the accuracy of the model. To implement our decision tree, we used the longitude and latitude data associated with each image to further classify each image. After, we fed the AlexNet predictions and the images into the decision tree. This improved our accuracy to 100%. The overall accuracy of our model from last year's model improved by 26.5%. We were able to reach our goal of achieving accurate polar low classification.

Future Directions

For furthering our research, future FIRE researchers can look into increasing the number of images that are used in the training and testing of our model. Increasing the images would increase confidence in our model's accuracy rate upon testing the images with our model. A group could also look into data-dropping certain images to further increase the accuracy of our model if the accuracy drops. Data-dropping is a technique involving randomly selected data points that are ignored during training. This could potentially solve the issue of overfitting.

Literature Cited

1. Terpstra, A., & Watanabe, S.-ichi. (2020, June 30). *Polar Lows.* Oxford Research Encyclopedia of Climate Science. Retrieved November 8, 2022, from https://doi.org/10.1093/acrefore/9780190228620.013.775
2. Bourke, D. (n.d.). *Pytorch Deep Learning.* GitHub. Retrieved November 2, 2022, from https://github.com/mrdbourke/pytorch-deep-learning/blob/main/06_pytorch_transfer_learning.ipynb